

CSC309 Programming on the Web

week 9: event loop revisit, jsonp

Amir H. Chinnai, Spring 2017

Office Hours: M 3:45-5:45 BA4222

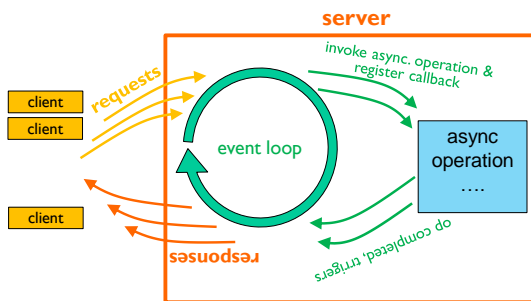
ahchinaei@cs.toronto.edu
<http://www.cs.toronto.edu/~ahchinaei/>

motivation

- ❖ **understanding event loop**
 - a couple of examples in week 6
 - more details (and live-coding) this week
- ❖ **cross-origin resource sharing**
 - requesting data from other domains
 - we saw http, XMLHttpRequest (XHR) already
 - jsonp this week

review 9-2

event loop



review 9-3

case study

```
// assume
// req1 is non-blocking: async part: ~4 s ; rest: ~0 s.
// req2 is synchronous: requires ~6 s.
// req3 is non-blocking: async part: ~2 s ; rest: ~0 s.
// req4 is synchronous: requires ~0 s.
```

```
// Question 1: req2 and req4 enter at time 12, in order;
// what time is each responded?
```

```
// Question 2: req4 and req2 enter at time 12, in order;
// what time is each responded?
```

case study 9-4

case study

```
// req1 is non-blocking: async part: ~4 s ; rest: ~0 s.
// req2 is synchronous: requires ~6 s.
// req3 is non-blocking: async part: ~2 s ; rest: ~0 s.
// req4 is synchronous: requires ~0 s.
```

```
// Question 3: req3, req4, req1, and req2 enter at time 12;
// what time is each responded?
```

```
// Question 4: req3, req4, req2, and req1 enter at time 12;
// what time is each responded?
```

```
// Question 5: req2, req3, and req4 enter at time 12;
// what time is each responded?
```

case study 9-5

case study

```
// this simulates request 1, with an asynchronous function
$("#req1").click(function(){
    $("#21").html(" request 1 started at "+time());

    setTimeout(function timer() {
        $("#21").append(" and processed at "+time());
    }, 4000);

});
```

case study 9-6

case study

```
// this simulates request 2, a blocking one
$("#req2").click(function(){
  $("#22").html(" request 2 started at "+time());
  for( var k = 0; k<200000; k++) {
    console.log(k);
  }
  function x() {
    $("#22").append(" and processed at "+time());
  }
  x = x();
});
```

case study 9-7

case study

```
// this simulates request 3, with an asynchronous function
$("#req3").html(" request 3 started at "+time());

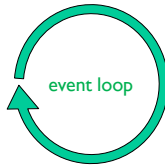
setTimeout (function timer() {
  $("#23").append(" and processed at "+time());
}, 2000);

// this simulates request 4
$("#req4").html(" request 4 processed at "+time());
```

case study 9-8

Question 3:

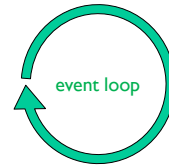
req3, req4, req1, and req2 enter at time 12;



case study 9-9

Question 4:

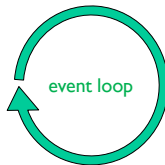
req3, req4, req2, and req1 enter at time 12;



case study 9-10

Question 5:

req2, req3, and req4 enter at time 12;



case study 9-11

jsonp

- ❖ **jsonp** with **padding**
 - application: requesting data from different domains
- ❖ recall:
 - sure the client can send an http **request** (or xhr),
 - to receive data from its server domain
- but,
- the **response** in the best form is some data **json** format

jsonp 9-12

example

- ❖ **request**
www.peopledb.com/first
- ❖ **response**
`{ "name": "John", "age": 30, "city": "New York" }`
- ❖ so, running a script like this

```
<script src="www.peopledb.com/first">
</script>
```


results in an error.

jsonp 9-13

jsonp

- ❖ you can wrap **json response**, the way you wish:
- ❖ **example: response**
`cb({ "name": "John", "age": 30, "city": "New York" })`
- ❖ now, the script looks like this to the browser:

```
<script src= cb({ "name": "John", "age": 30, "city": "New York" })>
</script>
```
- ❖ when the **request** sent is
www.peopledb.com/first?callback=cb

jsonp 9-14

informal feedback

- ❖ 75% rated assignments/proj the most effective in learning
- ❖ 58% rated kahoots effective in learning
- ❖ 52% stated more clear step-by-step lab instructions
- ❖ 40% stated more live-coding in class
- ❖ ~ less technologies to be covered
- ❖ ~ less assignments/projects
- ❖ peer evaluation can always be revised
- ❖ peer instruction a ramp-up approach

feedback 9-15

recall

what would you need to do well?

- ❖ pay attention to concepts (in **lectures**)
- ❖ practice the concepts and skills (in **labs**)
- ❖ master your skills by **assignments**
- ❖ put all your learning together in the **project**
- ❖ **start early** the assignments and project phases
- ❖ lectures and labs are limited
 - but for your deep learning, **sky's is the limit**
- ❖ **final exam**: deep concepts

Introduction 1-9

- spiral model

feedback 9-16

9-17