selsoft
build better, deliver faster

Developing RESTful Web Services with Java

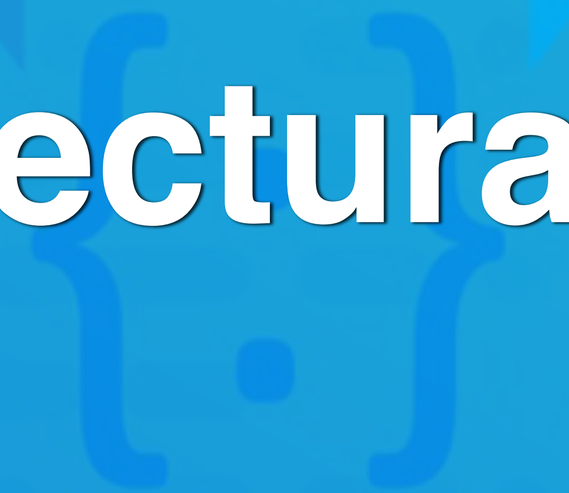
Chapter 5: RESTful As An Architectural Style

Eğitmen:

Akın Kaldıroğlu

Çevik Yazılım Geliştirme ve Java Uzmanı

RESTful As An Architectural Style



selsoft

build better, deliver faster

RESTful

- **REST** stands for **REpresentational State Transfer**.
- It is introduced by Roy Fielding's in his doctoral thesis *Architectural Styles and the Design of Network-based Software Architecture* in 2000.
- In his thesis he mainly made a research regarding how to apply the architecture of web to applications.
- And he developed an architectural style that he called **REST**.

Architectural Style

- Roy Fielding defines **architectural style** in his doctoral thesis as follows:

  An architectural style is a coordinated set of architectural constraints that restricts the roles/features of architectural elements and the allowed relationships among those elements within any architecture that conforms to that style.

Constraints

- Roy Fielding later lists following constraints on REST architectural style:
  - Client-server
  - Stateless
  - Cache
  - Uniform Interface
  - Layered System
  - Code on Demand, which is optional.

Resource and Representation - I

- The key abstraction of information in REST is a resource.
- REST uses a resource identifier to identify the particular resource involved in an interaction.
- REST components perform actions on a resource by using a representation to capture the current or intended state of that resource and transferring that representation between components.
- A representation is a sequence of bytes, plus representation metadata to describe those bytes.

Resource and Representation - II

- A given representation may indicate the current state of the requested resource, the desired state for the requested resource, or the value of some other resource, such as a representation of the input data within a client's query form, or a representation of some error condition for a response.
- The data format of a representation is known as a media type.

Stateless - I

- All REST interactions are stateless.
- That is, each request contains all of the information necessary for a connector to understand the request, independent of any requests that may have preceded it.
  - A connector is one of server, client, cache, etc.

Stateless - II

- This restriction accomplishes four functions:
  - No need for connectors to retain application state between requests, thus reducing consumption of physical resources and improving scalability;
  - Allowing parallel processing without requiring that the processing mechanism understand the interaction semantics;
  - Allowing an intermediary to view and understand a request in isolation
  - It forces all of the information to be present in each request.

Uniform Interface - I

- About uniform interface Fielding says, REST is defined by four interface constraints:
  - identification of resources;
  - manipulation of resources through representations;
  - self-descriptive messages; and,
  - hypermedia as the engine of application state.

Uniform Interface - II

- REST is resource-based.
- Each resource is idendified by a URI.
- For example if a **`GET`** with an id is sent to a conversion URI then a conversion with given id is returned.

```
@Path("greetings")
public class GreetingResource {
   …
   @GET
   @Path("{language}")
   public String getGreeting(…) {…}
}
```

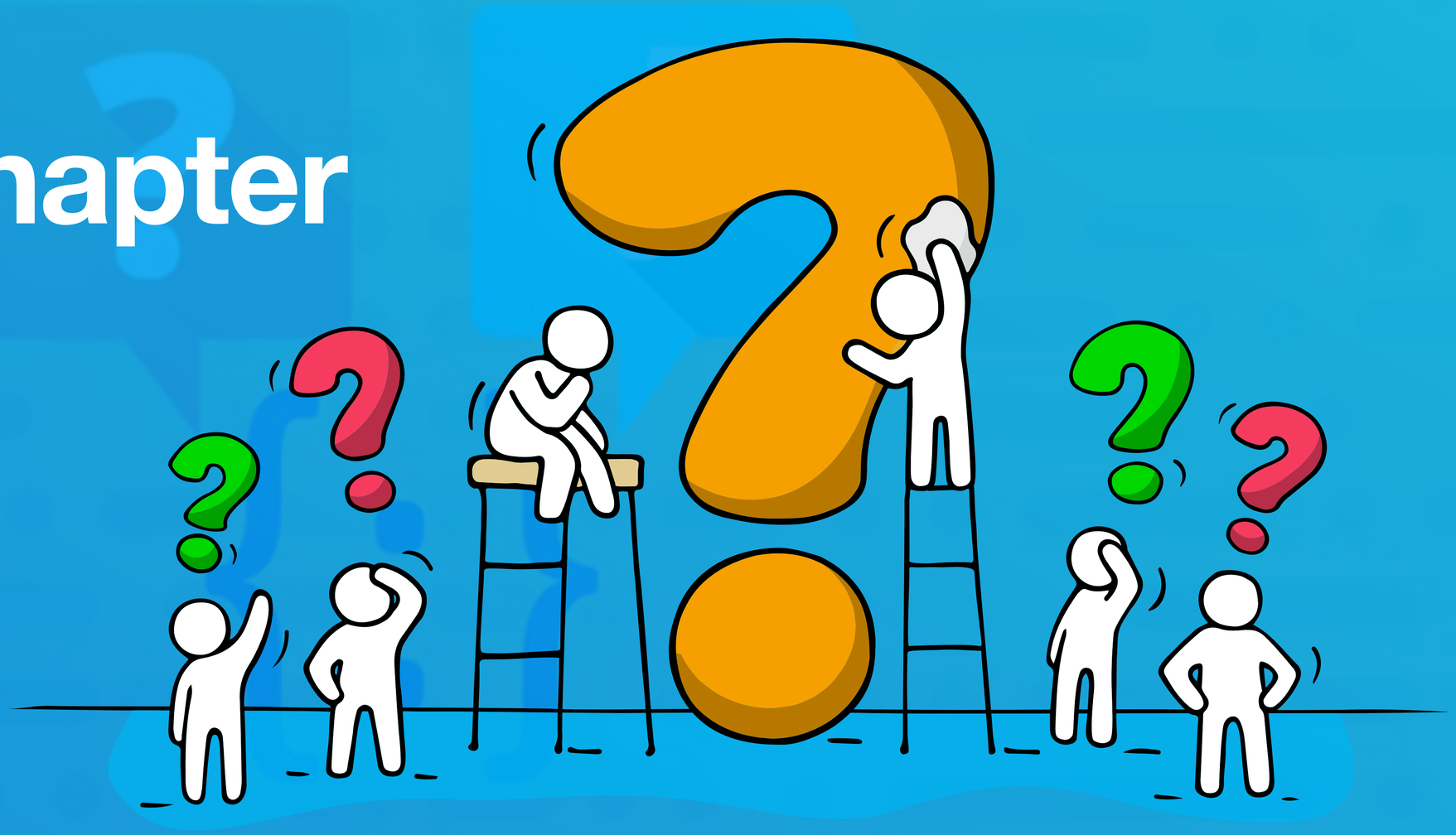
```
@Path("greetings")
public class GreetingResource {
   …
   @POST
   @Path("{language}/{greeting}")
   public Response createGreeting(…)
}
```

Uniform Interface - III

- What is transfered to the client is a representation of the conversion resource.
- But whether the representation is in the same format as the raw source, or is derived from the source, remains hidden behind the interface.

End of Chapter

Time for Questions!



selsoft
build better, deliver faster

info@selsoft.com.tr
selsoft.com.tr