

JAX-RS API FOR RESTFUL WEB SERVICES

40

40

JAX-RS

- RESTful Web services implemented as methods of objects
- `@Path` for class: base context root
- `@Path` for methods: extensions for resources
- `@Get`, `@Post`, `@Put`, etc for methods
- `@Produces`, `@Consumes`: MIME types
- `@QueryParam`: param from query string

41

41

Example

```
@Path("/hello")
public class HelloResource {
    @GET
    @Produces("text/plain")
    public String sayHello (
        @QueryParam("name")
        String name ) {
        return "Hello, " + name;
    }
}
```

GET http://host/HelloService?name=Joe

42

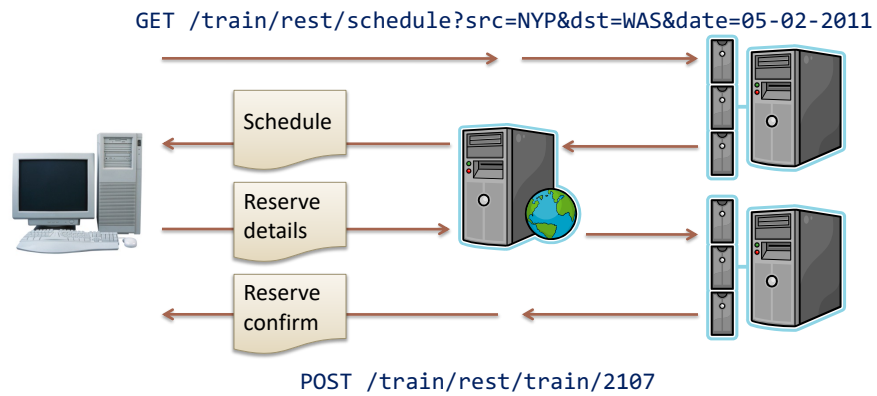
42

EXAMPLE: TRAIN RESERVATION SERVICE

43

43

REST Web Service



44

Application Class

```
@ApplicationPath("/train/rest")
public class TrainApp extends Application {
    public Set<Class<?>> getClasses() {
        Set<Class<?>> s = new HashSet<Class<?>>();
        s.add(ScheduleResource.class);
        s.add(TrainResource.class);
        return s;
    }
}
```

GET /train/rest/schedule?src=NYP&dst=WAS&date=05-02-2011

45

45

Schedule Resource

```
@Path("/schedule")
public class ScheduleResource {

    ...

    @GET
    @Produces("application/vnd.trains+json")
    public Response getSchedule (
        @QueryParam("src")
        String start,
        @QueryParam("dst")
        String destination,
        @DefaultValue("today")
        @QueryParam("date")
        String travelDate)
```

GET

/train/rest/schedule?src=src&dest=dest&date=travDate

46

RESOURCES AND REPRESENTATIONS

47

47

```

@Path("/schedule")
public class ScheduleResource {
    @Context UriInfo uriInfo;

    ... ISchedule scheduleService;

    @GET
    @Produces("application/vnd.trains+json")
    public Response getSchedule (
        @QueryParam("src") String start,
        @QueryParam("dst") String destination,
        @DefaultValue("today")
        @QueryParam("date") String travelDate)
    {
        ScheduleRepresentation schedule =
            scheduleService.get(start, destination,
                travelDate, uriInfo);
        return Response.ok(schedule)...build();
    } } }

```

48

48

Representation Class

```

public abstract class Representation {
    public static final String
        RELATIONS = "http://www.example.org"
            + "/train/rest/relations/";
    public static final String
        RELATION_RESERVE = RELATIONS + "reserve";
    public static final String
        RELATION_CANCEL = RELATIONS + "cancel";

    public static final String
        MEDIA_TYPE = "application/vnd.trains+json";

    public abstract List<Link> getLinks();
}

```

49

49

Representation Class

```
public abstract class Representation {
    public static final String
        RELATIONS = "http://www.example.org"
                    + "/train/rest/relations/";
    public state final String
        RELATION_RESERVE = RELATIONS + "reserve";
    public state final String
        RELATION_CANCEL = RELATIONS + "cancel";

    public static final String
        MEDIA_TYPE = "application/vnd.trains+json";

    public abstract List<Link> getLinks();
}

Link: <http://www.example.org/train/rest/train/2103>;
rel="http://www.example.org/train/rest/relations/reserve";
type="application/vnd.trains+json"
```

50

Representation Class

```
public abstract class Representation {
    public static final String
        RELATIONS = "http://www.example.org"
                    + "/train/rest/relations/";
    public state final String
        RELATION_RESERVE = RELATIONS + "reserve";
    public state final String
        RELATION_CANCEL = RELATIONS + "cancel";

    public static final String
        MEDIA_TYPE = "application/vnd.trains+json";

    public abstract List<Link> getLinks();
}

Link: <http://www.example.org/train/rest/train/2103>;
rel="http://www.example.org/train/rest/relations/reserve";
type="application/vnd.trains+json"
```

51

Representation Class

```
public abstract class Representation {
    public static final String
        RELATIONS = "http://www.example.org"
                    + "/train/rest/relations/";
    public static final String
        RELATION_RESERVE = RELATIONS + "reserve";
    public static final String
        RELATION_CANCEL = RELATIONS + "cancel";

    public static final String
        MEDIA_TYPE = "application/vnd.trains+json";

    public abstract List<Link> getLinks();
}

Link: <http://www.example.org/train/rest/train/2103>;
rel="http://www.example.org/train/rest/relations/reserve";
type="application/vnd.trains+json"
```

52

Train DTO

```
public class TrainDto {

    public long id;

    public String time;

    @Exclude
    public Link link;

    public TrainDto (Train t, UriBuilder ub) {
        this.id = t.getId();
        this.time = t.getTime();
        this.link = Link.fromUri(ub.build(id))
                        .rel(RELATION_RESERVE),
                        .type(MEDIA_TYPE)
                        .build();
    }
}
```

53

53

Schedule Representation

```
public class ScheduleRepresentation extends Representation {

    public List<TrainDto> trains;

    public ScheduleRepresentation (List<Train> ts,
                                   UriBuilder ub) {
        trains = new ArrayList<TrainDto>();
        for (Train t : ts)
            trains.add(new TrainDto(t,ub));
    }

    public List<Link> getLinks() {
        List<Link> links = new ArrayList<Link>();
        for (TrainDto t : trains) links.add(t.link);
        return links;
    }
}
```

54

54

```
@Path("/schedule")
public class ScheduleResource {
    @Context UriInfo uriInfo;

    ... ISchedule scheduleService;

    @GET
    @Produces("application/vnd.trains+json")
    public Response getSchedule (...)
    {
        ScheduleRepresentation schedule =
            scheduleService.get(start, destination,
                               travelDate, uriInfo);
        Response.Builder response = Response.ok(schedule);
        for (Link trainLink : schedule.getLinks())
            response = response.link(trainLink);
        return response.build();
    } } }
```

55

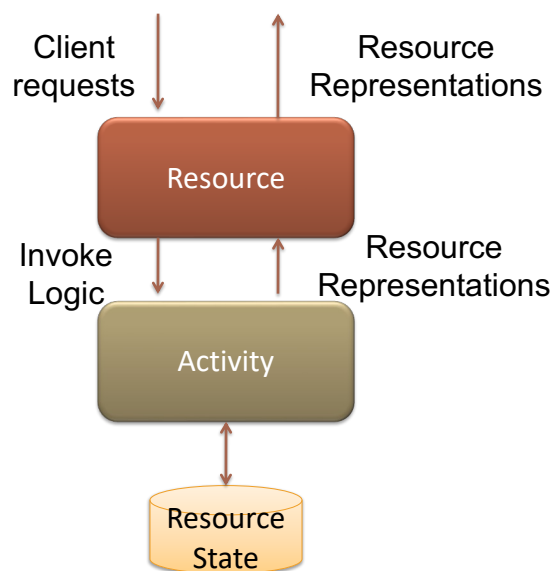
55

ADDING BUSINESS LOGIC

56

56

Business Activities



57

57

```

@Path("/schedule")
public class ScheduleResource {

    @Context UriInfo uriInfo;

    ISchedule scheduleService = new ScheduleActivity();

    @GET
    @Produces(ScheduleRepresentation.MEDIA_TYPE)
    public Response getSchedule (
        @QueryParam("src") String start,
        @QueryParam("dst") String destination,
        @DefaultValue("today")
        @QueryParam("date") String travelDate) {
        ScheduleRepresentation schedule =
            scheduleService.get(start, destination,
                               travelDate, uriInfo);
        return Response.ok(schedule)...build();
    }
}

```

58

58

```

@Path("/schedule")
public class ScheduleResource {

    @Context UriInfo uriInfo;

    @Inject ISchedule scheduleService

    @GET
    @Produces(ScheduleRepresentation.MEDIA_TYPE)
    public Response getSchedule (
        @QueryParam("src") String start,
        @QueryParam("dst") String destination,
        @DefaultValue("today")
        @QueryParam("date") String travelDate) {
        ScheduleRepresentation schedule =
            scheduleService.get(start, destination,
                               travelDate, uriInfo);
        return Response.ok(schedule)...build();
    }
}

```

59

59

Schedule Activity

```
public interface ISchedule {
    public ScheduleRepresentation get (
        String start, String destination, String travelDate,
        UriInfo ui );
}

public class ScheduleActivity implements ISchedule {
    public ScheduleRepresentation get (
        String start, String destination, String travelDate,
        UriInfo ui ) {
        List<Train> trains = ...;
        UriBuilder ub = ui.getBaseUriBuilder();
        ub = ub.path("train").path("{tid}");
        return new ScheduleRepresentation (trains, ub);
    }
}
```

60

60

Schedule Activity

```
public interface ISchedule {
    public ScheduleRepresentation get (
        String start, String destination, String travelDate,
        UriInfo ui );
}

@RequestScoped
public class ScheduleActivity implements ISchedule {
    public ScheduleRepresentation get (
        String start, String destination, String travelDate,
        UriInfo ui ) {
        List<Train> trains = ...;
        UriBuilder ub = ui.getBaseUriBuilder();
        ub = ub.path("train").path("{tid}");
        return new ScheduleRepresentation (trains, ub);
    }
}
```

61

61

Train Resource

```
@Path("/train")
public class TrainResource {
    @Context UriInfo uriInfo;
    @Inject ITrain trainService;
    @POST
    @Path("/{tid}")
    public Response makeReservation (
        @PathParam("tid") String tid,
        @DefaultValue("business")
        @QueryParam("class")
        String _class) {
        URI reserveURI =
            trainService.reserve(tid, _class, uriInfo, _class);
        return Response.created(reserveURI).build();
    }
}
```

POST /train/rest/train/tid[?class=class]

62

62

Reservation Resource

```
@Path("/reservation")
public class ReservationResource {
    @Context UriInfo uriInfo;
    @Inject IReservation reservationService;

    @GET
    @PATH("/{rid}")
    @Produces(ReservationRepresentation.MEDIA_TYPE)
    public Response getReservation (
        @PathParam("rid") String rid ) {
        ReservationRepresentation reservation =
            reservationService.get(rid, uriInfo);
        return Response.ok(reservation)...build();
    }
}
```

GET /train/rest/reservation/rid

63

63

RESTful Shopping Cart

```
@Path("/rest/shoppingcart/{cartId}")
public class ShoppingCartResource {
    @PathParam("cartId") String cartId;
    ... EntityManager em;
    @PUT
    public Response newCart () {
        CartEntity cart = em.find(CartEntity.class, cartId);
        if (cart != null)
            throw new WebApplicationException
                (Response.Status.FORBIDDEN);
        cart = new CartEntity(cartId);
        em.persist(cart);
        return Response.status(Response.Status.CREATED);
    }
}
```

64

64