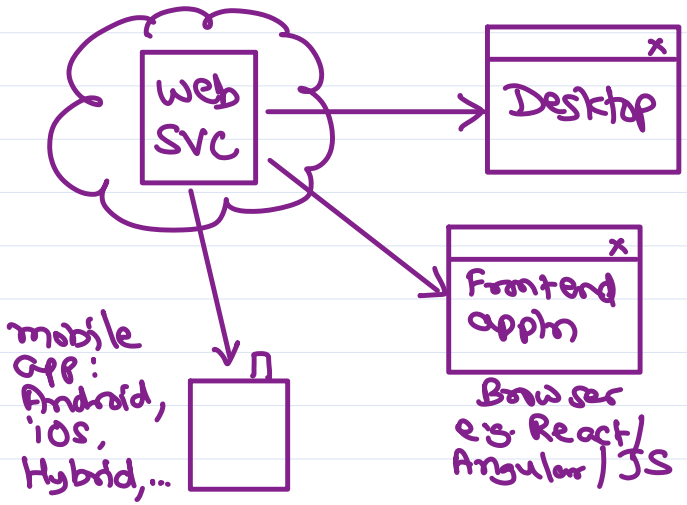
Advanced Java

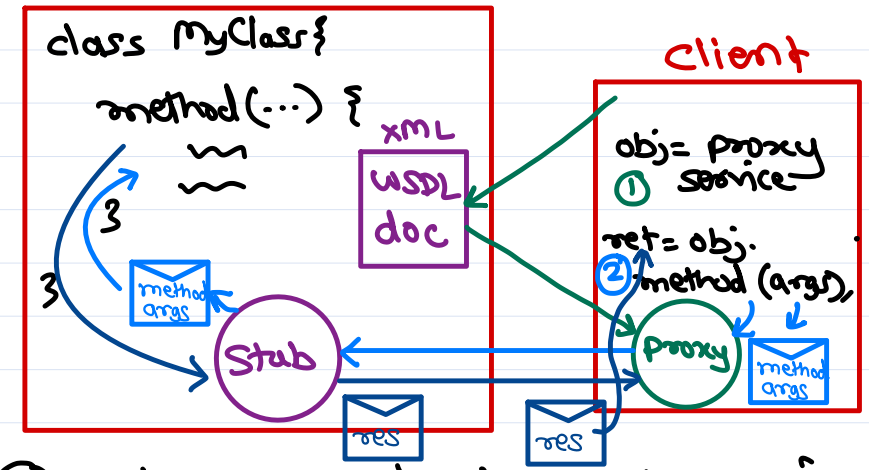
Trainer: Nilesh Ghule

Web Service

* No User Interface
  * Input : Req Body
  * Output: Response Body
* Protocol : http
* Data format: xml/JSON
* e.g. Currency Converter, Weather Service, Cricket Score, ...



SOAP/XML web services.

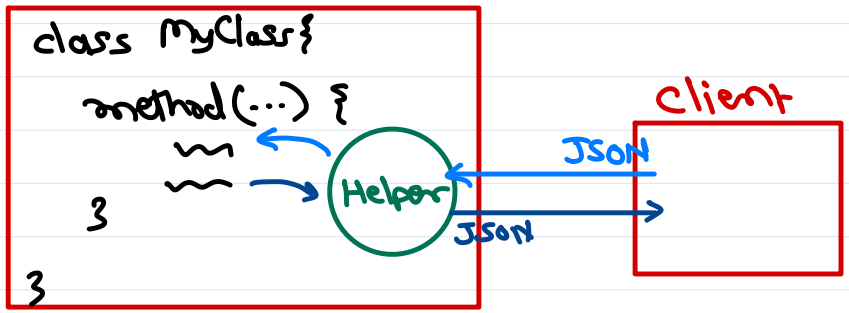


① web server hosts web service appln & stub is created at server side.
② WSDL doc read by client & create proxy obj at client side.
③ client calls required method on proxy, which inturn send method & arg details to server side stub.
④ stub calls actual method & send result back to client proxy.
⑤ proxy return result to client

REST services.

RE: Representational
S : State (object state).
T: Transfer

* Representation/Format: JSON/XML
* Protocol: http



* No validation of data (like WSDL) is done on client side, so calling REST services at client is very light weight / faster.

REST services

* Based on HTTP protocol.
* Use HTTP request methods.

① GET http://lh:8080/books → get all records
  * Request Body : No
  * Response Body : [{bk1}, {bk2}, ...]

  http://lh:8080/books/{id} → get record of given id (path variable)
  * Request Body : No
  * Response Body : {"id": .., "name": .., "author": .., "subject": .., "price": ..}

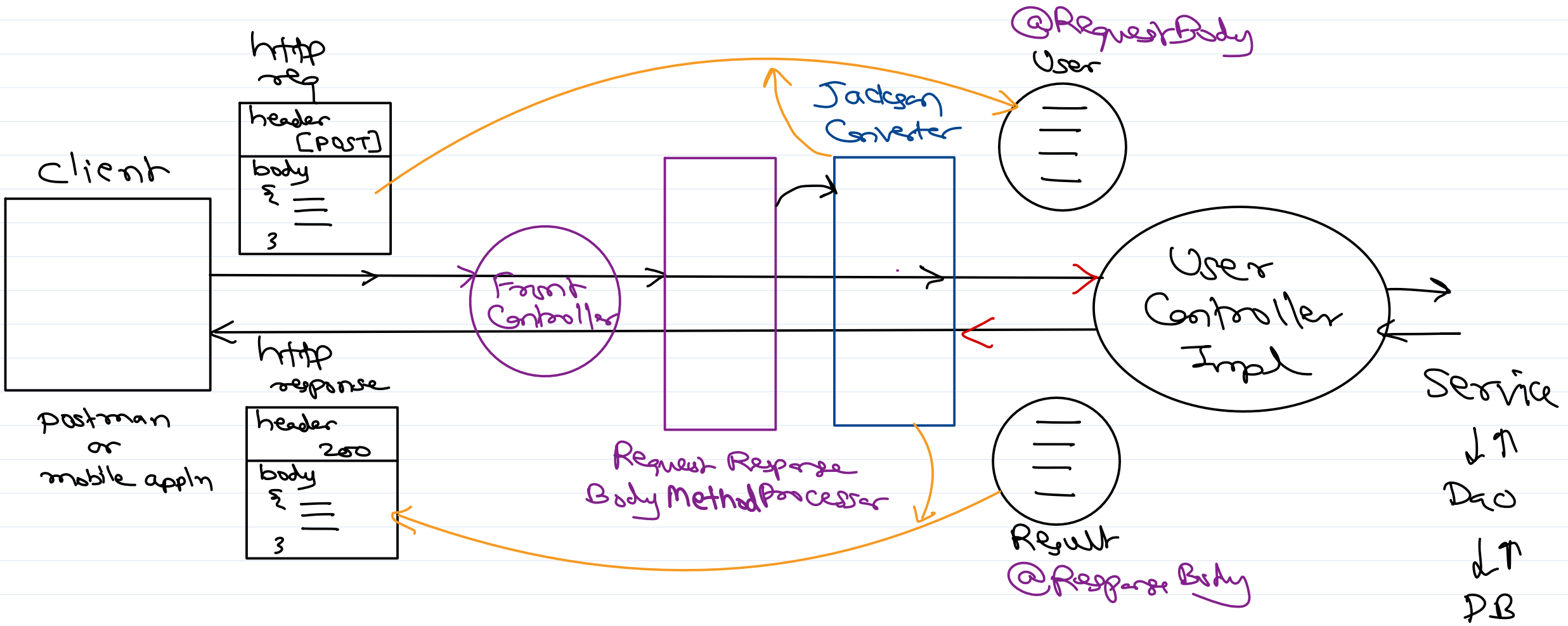
② POST http://lh:8080/books → create/send new record
  * Request Body : {"id": .., "name": .., "author": .., "subject": .., "price": ..}
  * Response Body : {"message": ..} → status: 201

③ DELETE http://lh:8080/books/{id} → delete record by id
  * Request Body : No
  * Response Body : {"message": ..}

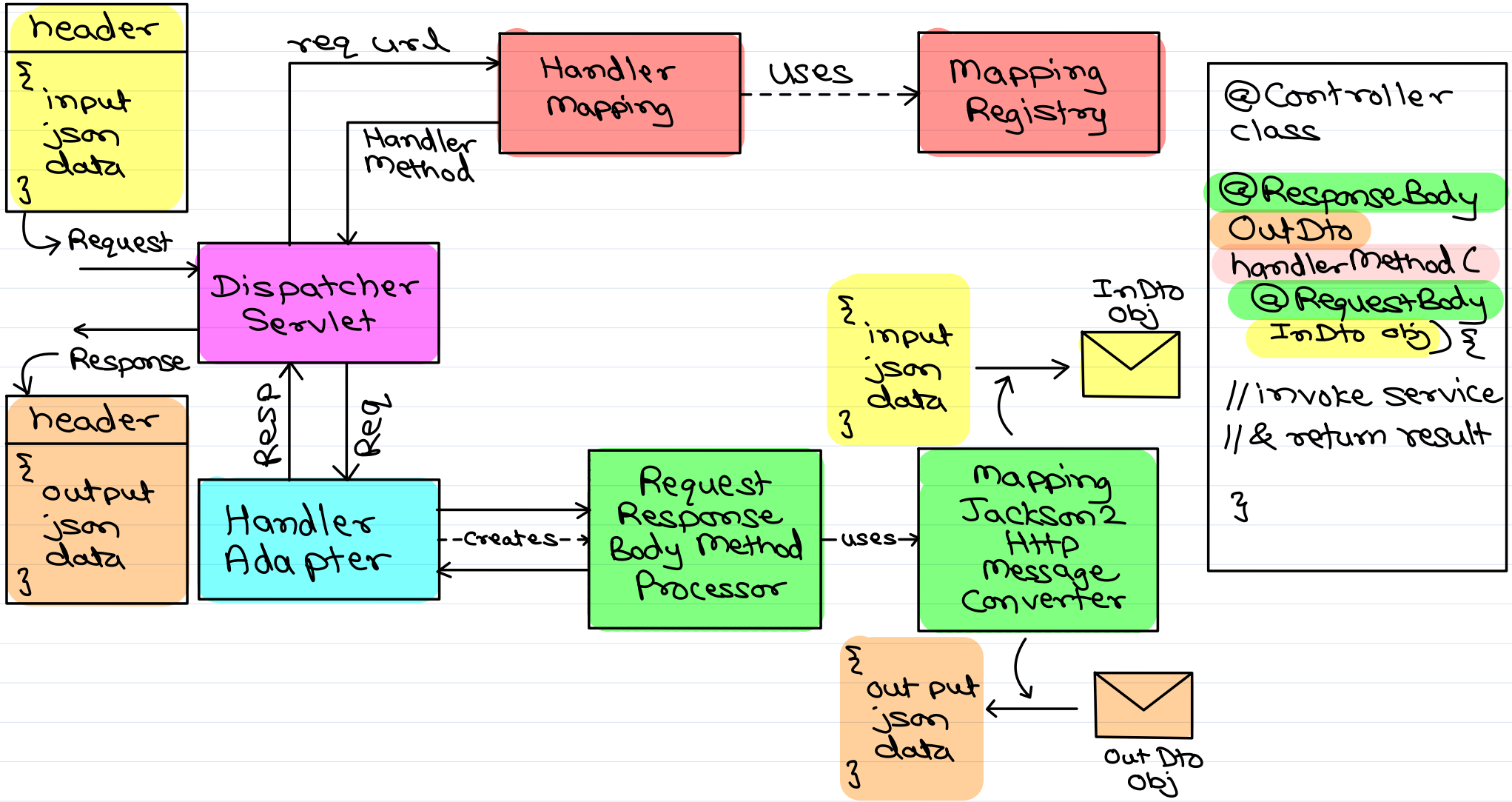
④ PUT http://lh:8080/books/{id} → modify whole record by id
  * Request Body : {~~"id": ..~~, "name": .., "author": .., "subject": .., "price": ..}
  * Response Body : {"message": ..}

⑤ PATCH http://lh:8080/books/{id} → modify partial record by id
  * Request Body : {"price": ..} * Response Body : {"message": ..}

REST Services



REST Services



Thank you!

Nilesh Ghule <nilesh@sunbeaminfo.com>