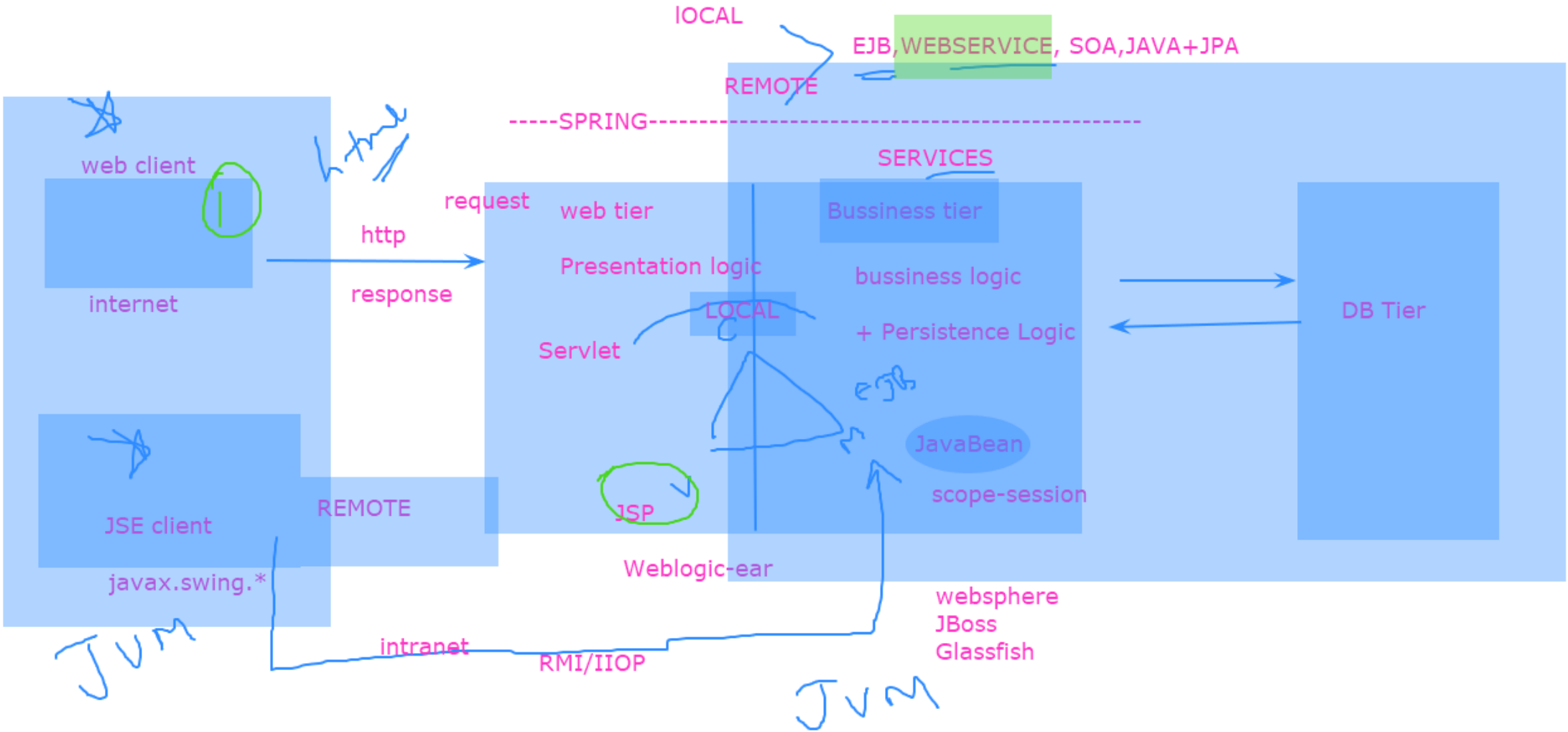
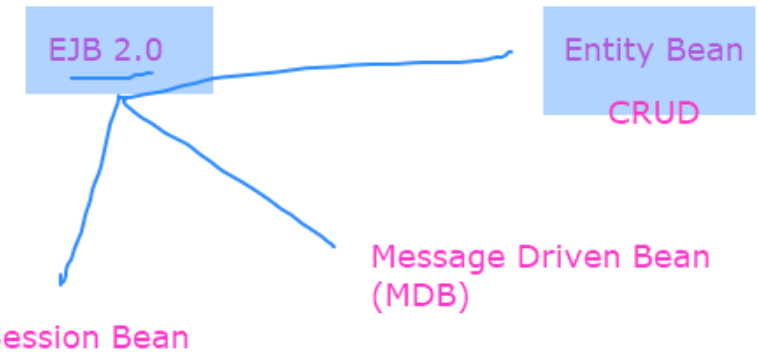

lOCAL
EJB,WEBSERVICE, SOA,JAVA+JPA
REMOTE
-----SPRING----------
SERVICES
web client
request
web tier
Bussiness tier
http
Presentation logic
bussiness logic
internet
response
LOCAL
+ Persistence Logic
DB Tier
Servlet
JavaBean
JSE client
REMOTE
JSP
scope-session
Weblogic-ear
javax.swing.*
websphere
JBoss
Glassfish
intranet
RMI/IIOP

jsp + SERVLETS + ejb



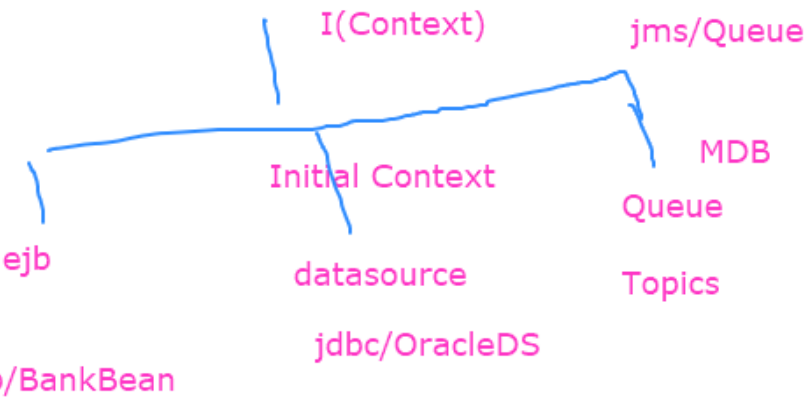
Bussiness Tier Design Patterns:

1. Session Facade
2. Service Locator (JNDI)
3. Bussiness Delegate

Lookup

JNDI

Java Naming and Directory Interface



Statless

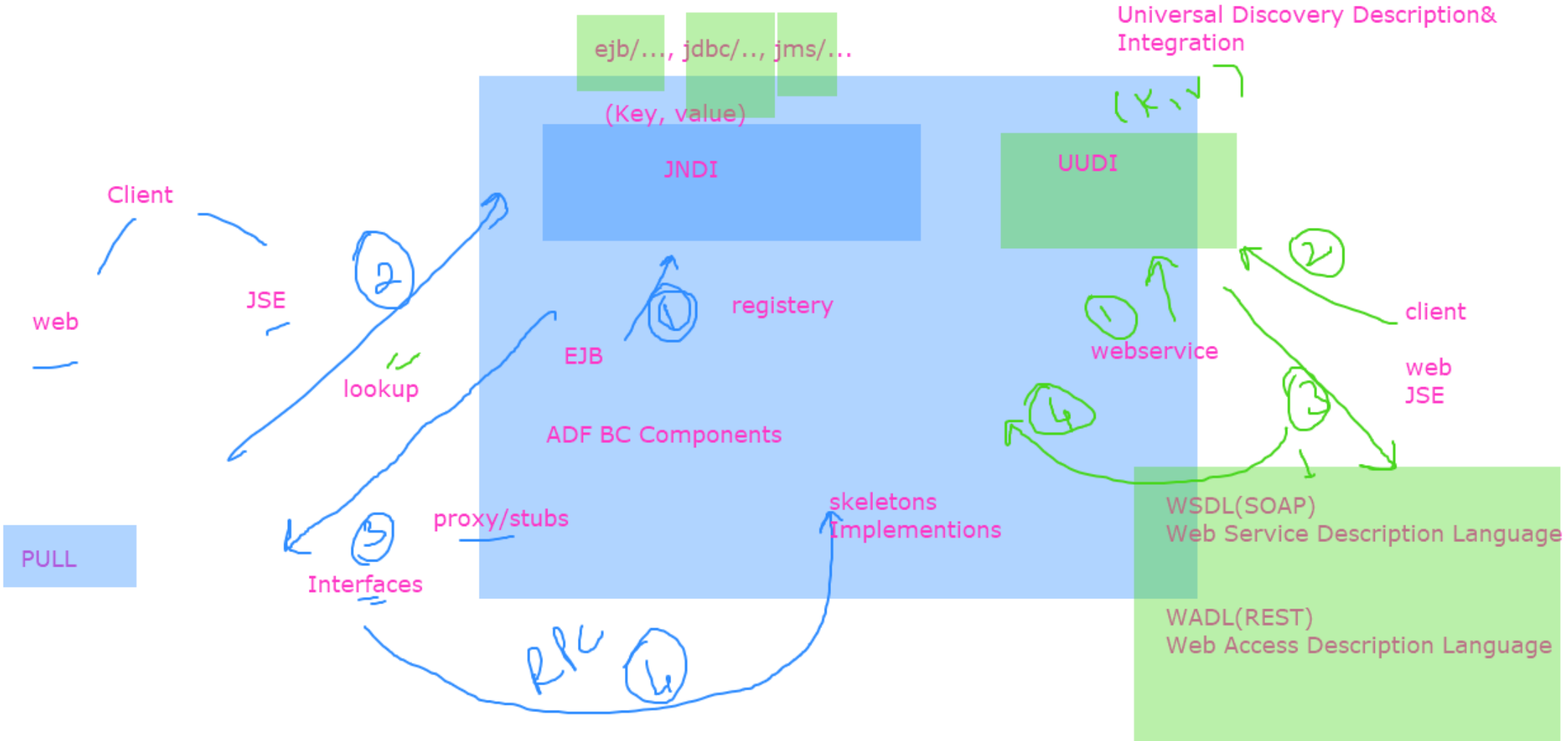
Stateful

client----Servlet---ejb(session Bean)---EntityBean----DB

JSP

JSF

sESSION fACADE


ejb/..., jdbc/.., jms/...
Universal Discovery Description&
Integration
(Key, value)
JNDI
UUDI
Client
web
JSE
lookup
EJB
registery
webservice
client
web
JSE
ADF BC Components
proxy/stubs
skeletons
Implementions
Interfaces
PULL
WSDL(SOAP)
Web Service Description Language
WADL(REST)
Web Access Description Language

EJB 2.0---Spring

Design Patterns: PUSH

IOC(Inversion of Control)/
Dependency Injection

EJB 3.*

Session Bean

Message Briven Bean

JPA (ORM)

JEE

JSE

AOP- Ascpect Oriented Programming

Logging

Transaction

Authn

multiple
code

Spring Core-JSE

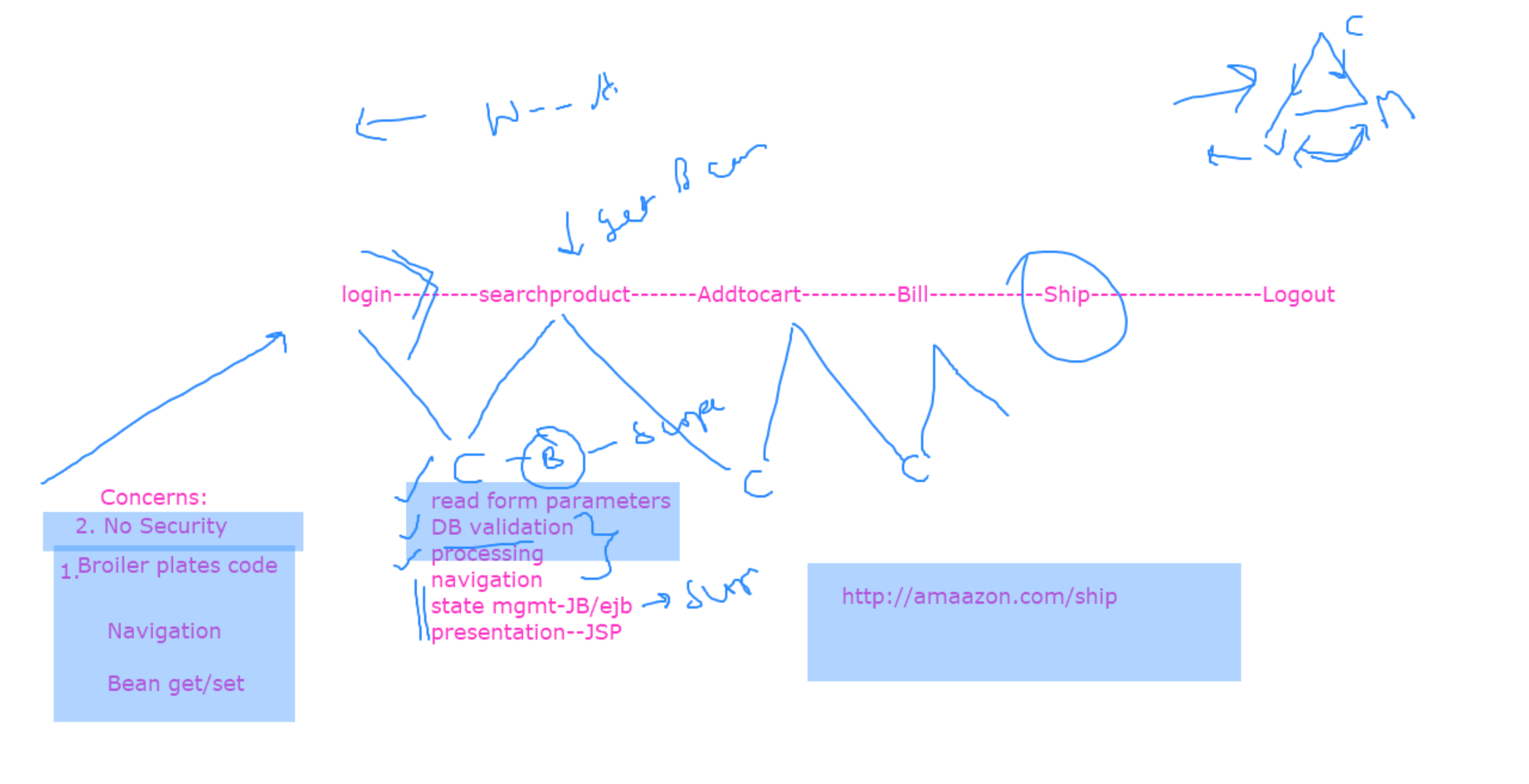
Spring MVC + JPA

Loose Coupling

Core Logic (POJO/JavaBean)

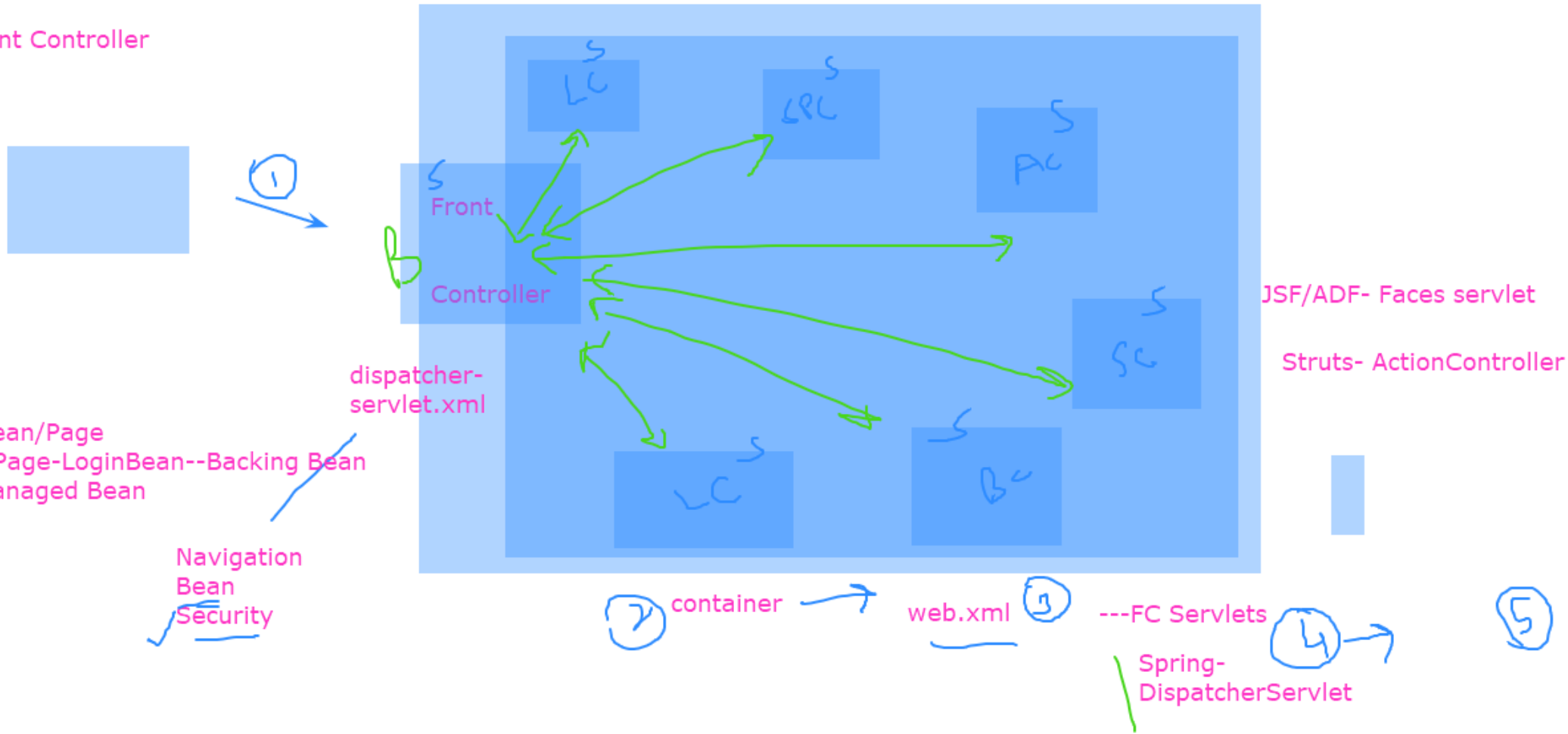
+

XML/Annotation


login----------searchproduct-------Addtocart----------Bill----------Ship----------Logout
Concerns:
2. No Security
1.Broiler plates code
Navigation
Bean get/set
read form parameters
DB validation
processing
navigation
state mgmt-JB/ejb
presentation--JSP
http://amaazon.com/ship

Presentation Tier:

Front Controller



WADL

REST - Respresentation State Transfer

Resource

http protocol -STATELESS

GET
POST
PUT
DELETE

FORMAT OF DATA:

JSON/XML/HTM/TEXT

Lightweight

WSDL

SOAP-Simple Access Object Protocol

Service

SOAP/HTTP- stateful

SOAPRequest/ SOAPResponse

XML

Heavyweight

Spring Boot:

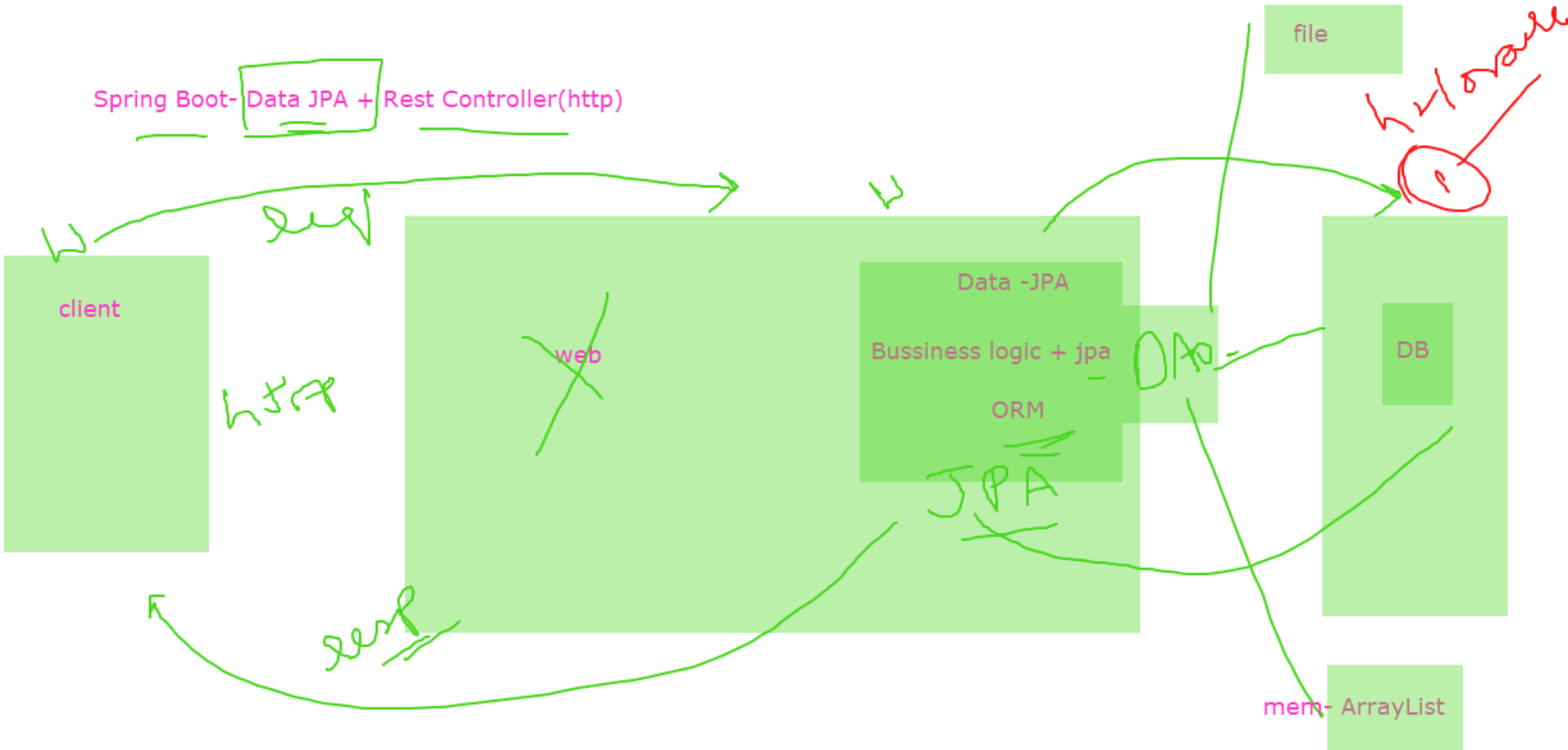
IDE:

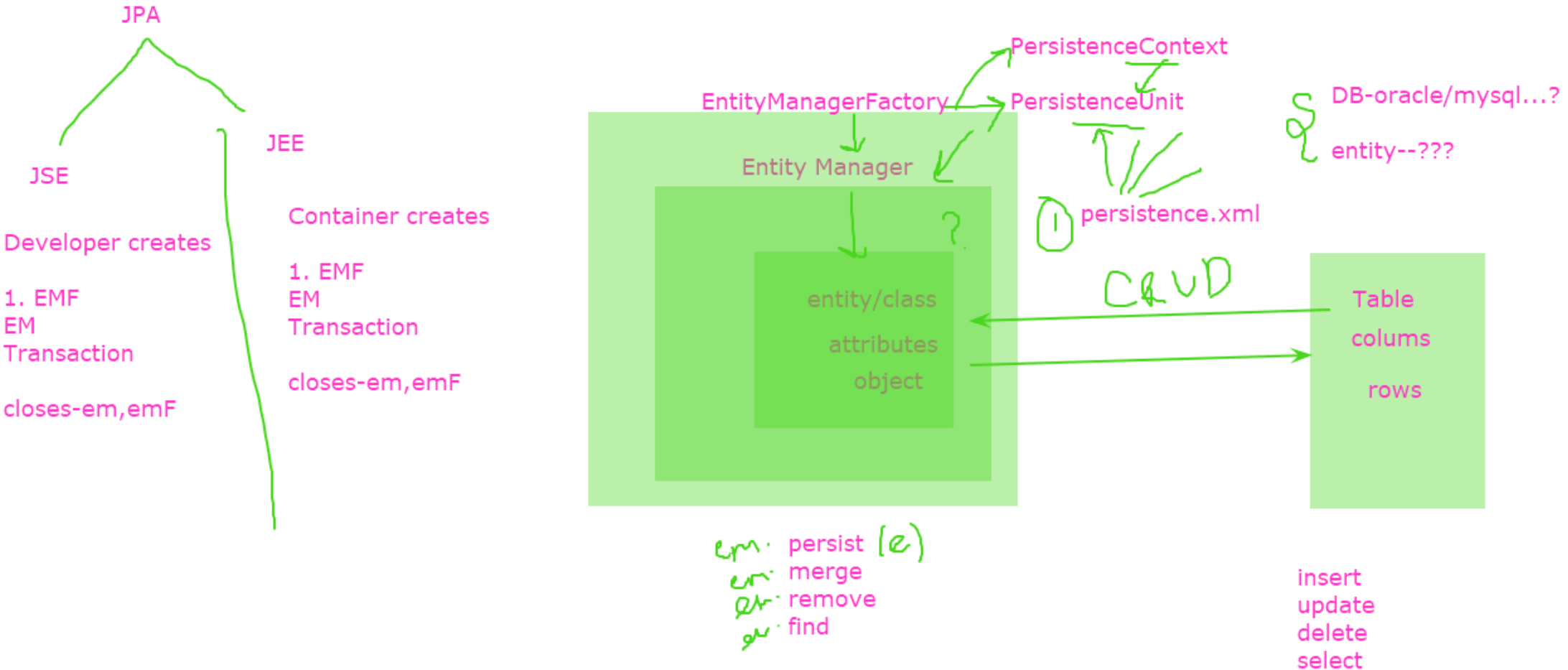
Eclipse(install plug in for STS)

STS- Sprint Tool Suite

Maven Projets

development
test
build
packagae
docume
deploy


Spring Boot- Data JPA + Rest Controller(http)
client
web
Data -JPA
Bussiness logic + jpa
ORM
JPA
DB
file
mem- ArrayList


JPA
JSE
JEE
Developer creates
1. EMF
EM
Transaction
closes-em,emF
Container creates
1. EMF
EM
Transaction
closes-em,emF
PersistenceContext
EntityManagerFactory
PersistenceUnit
Entity Manager
persistence.xml
DB-oracle/mysql...?
entity--???
entity/class
attributes
object
CRUD
Table
colums
rows
persist (e)
merge
remove
find
insert
update
delete
select

SpringBoot+Rest+Data-JPA

1. Configure the DB

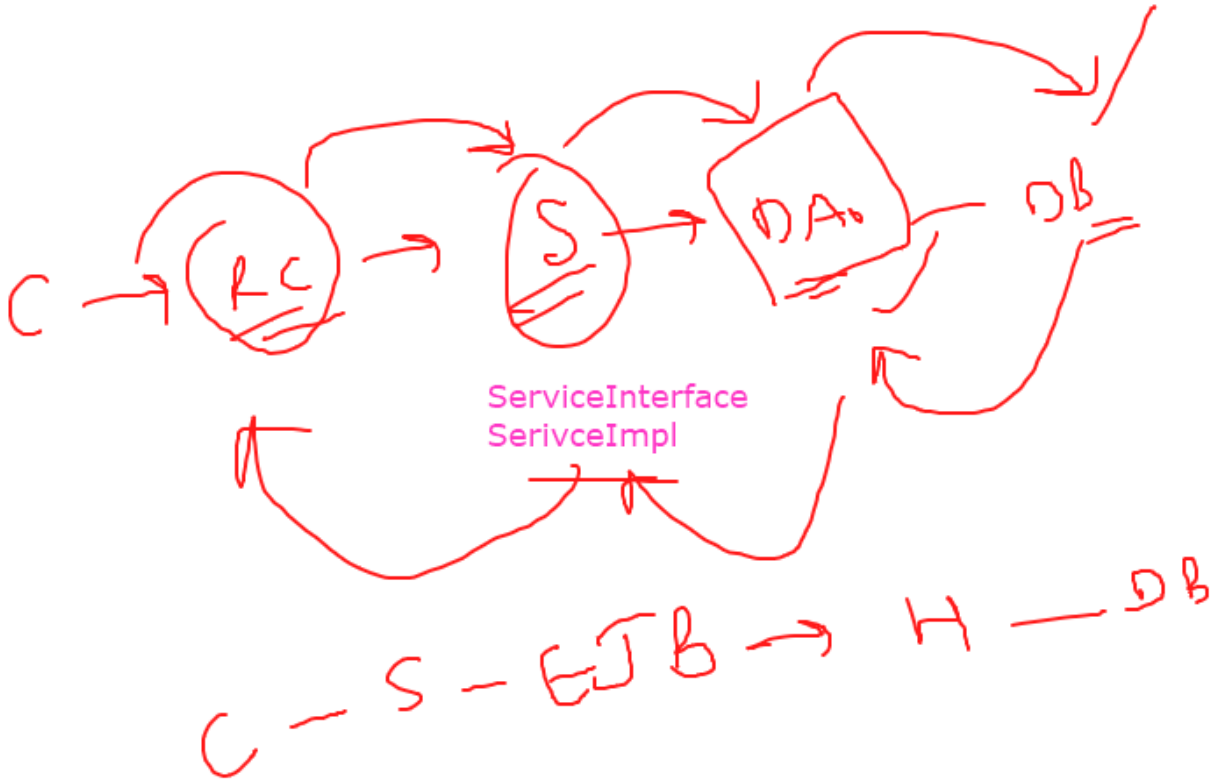
2. Enttiy==table

3. DAO-crud-DB

Jpa repository

4. Service-Bussissness Logic
Interface- method
Implementation-

5. Controller



SOA- Service Oriented Architecture

Adapeter /wrapper
Webservice

XML

Microservices

Microservice

Eureka Registery