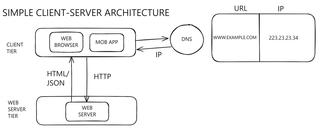
ANYTHING THAT WE WILL
DO IN HLD IS TO
MAKE OUR APPLICATION
SCALE BETTER

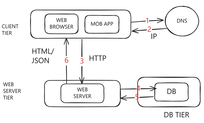
SIMPLE CLIENT-SERVER ARCHITECTURE



HTTP :

POST, GET, PUT ETC
URL : WWW.EXAMPLE.COM
HEADERS:
BODY : {
"USERID" : 1,
"NAME" : "ADITYA";
}

INTRODUCED DB LAYER



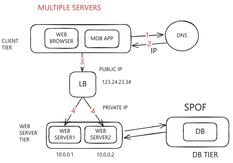
WHICH DB TO USE?
RELATIONAL : SQL BASED : MYSQL, POSTGRESQL, ORACLE SQL

NON - RELATIONAL

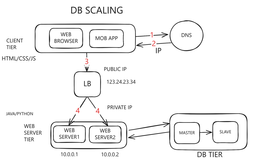
KEY - VALUE STORE : AMAZON DYNAMO DB
COLUMN BASED STORE : CASANDRA DB
DOCUMENT BASED STORE : MONGO DB
GRAPH STORE : INSTAGRAM, FB, TWITTER ETC

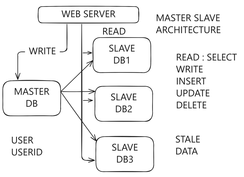
EVERY DATA CANT BE STRUCTURED

MULTIPLE SERVERS

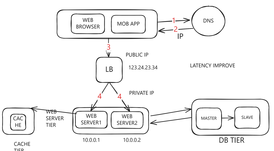


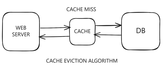
DB SCALING





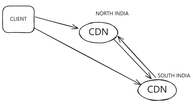
INTRODUCES CACHE





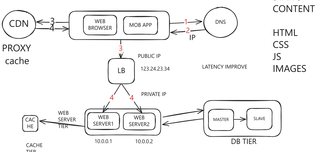
CONTENT DELIVERY NETWORK

AMAZON : CLOUDFRONT
AKAMAI



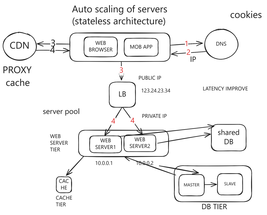
STATIC CONTENT

HTML
CSS
JS
IMAGES



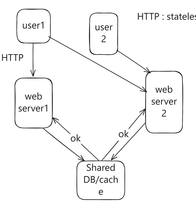
Auto scaling of servers
(stateless architecture)

cookies

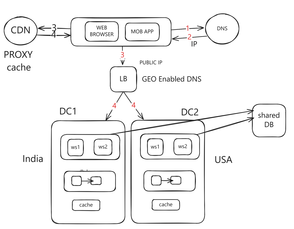


sticky sessions : stateful architecture

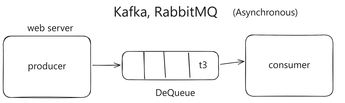
stateless architecture



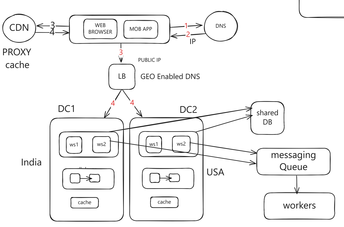
Multiple Data centres



Kafka, RabbitMQ (Asynchronous)

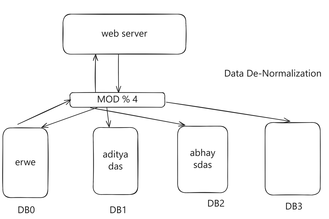


MESSAGING QUEUE



PUB-SUB (publisher - subscriber model)

SHARDING (DB partioning) Horizontal scaling



Data De-Normalization