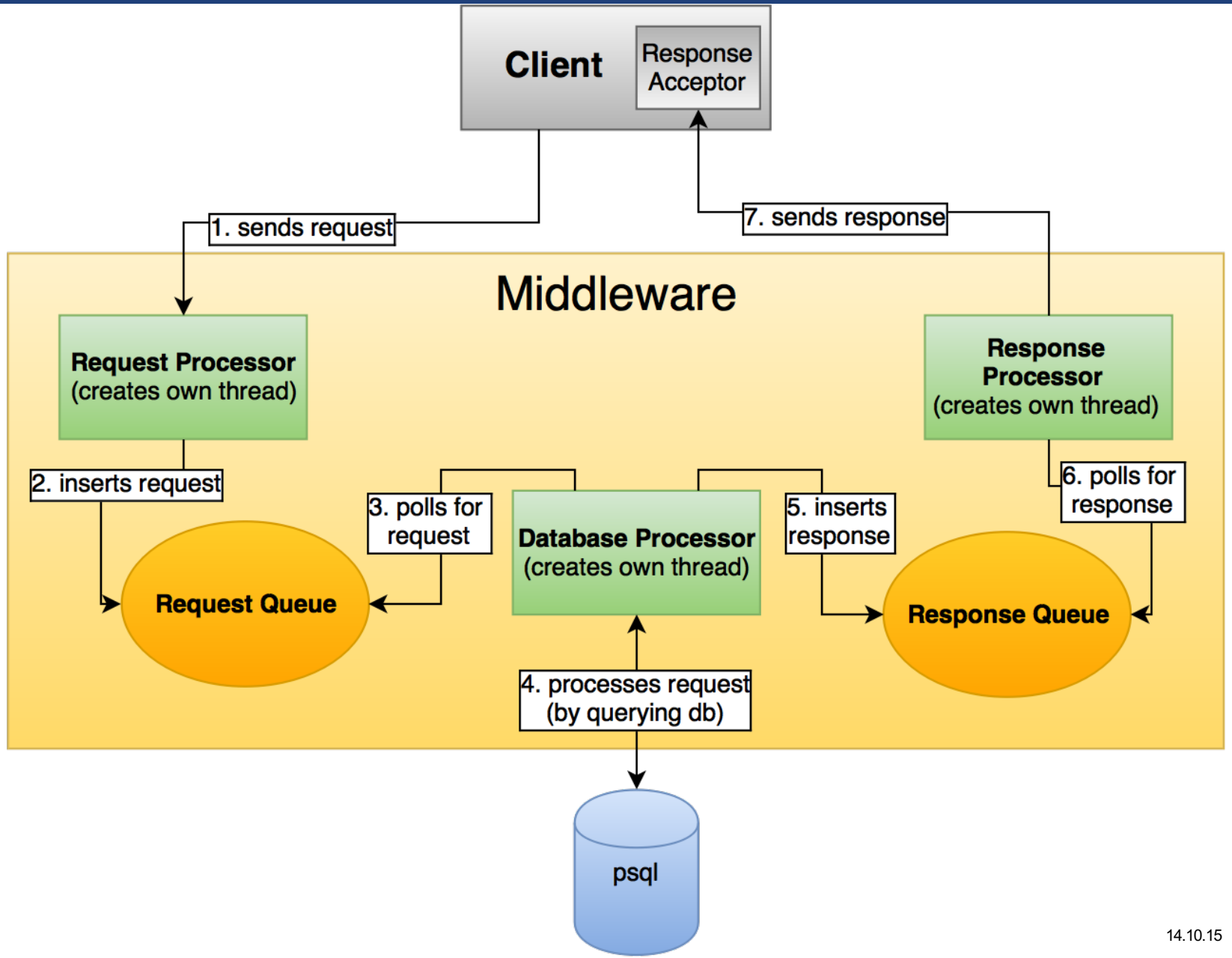
ETH zürich

Advanced Systems Lab
System Architecture

Dina Zverinski


Client
Response Acceptor
1. sends request
7. sends response
Middleware
Request Processor (creates own thread)
Response Processor (creates own thread)
2. inserts request
3. polls for request
Database Processor (creates own thread)
5. inserts response
6. polls for response
Request Queue
Response Queue
4. processes request (by querying db)
psql

Remarks

- Client has an own thread with a ServerSocket to be able to accept responses anytime
- Client disconnects from Middleware as soon as a request is sent
- Client only sends next request after receiving a response (callback), Middleware always has to respond
- Client has to register first, in order to get an ID (but will be assigned the same ID if he reregisters)
- Middleware is consists of 3 main threads (processors), that are hold an own thread pool of “workers”

Database

- Client: ID, IP Address, Port
- Queue: ID
- Message: ID, sender ID, receiver ID, queue ID, content, timestamp

- IDs are foreign keys (to make sure a valid ID is inserted)
- IP & Port needed to allow Client to reregister

- Connection pool at Database Processor

Communication

- Strings
- Start with “Codes” (for any kind of request/response/error)
- Comma separated
- Parsed accordingly by Client and Middleware

Thank you for your attention.