

# Managing database connections in REST APIs

database

connection

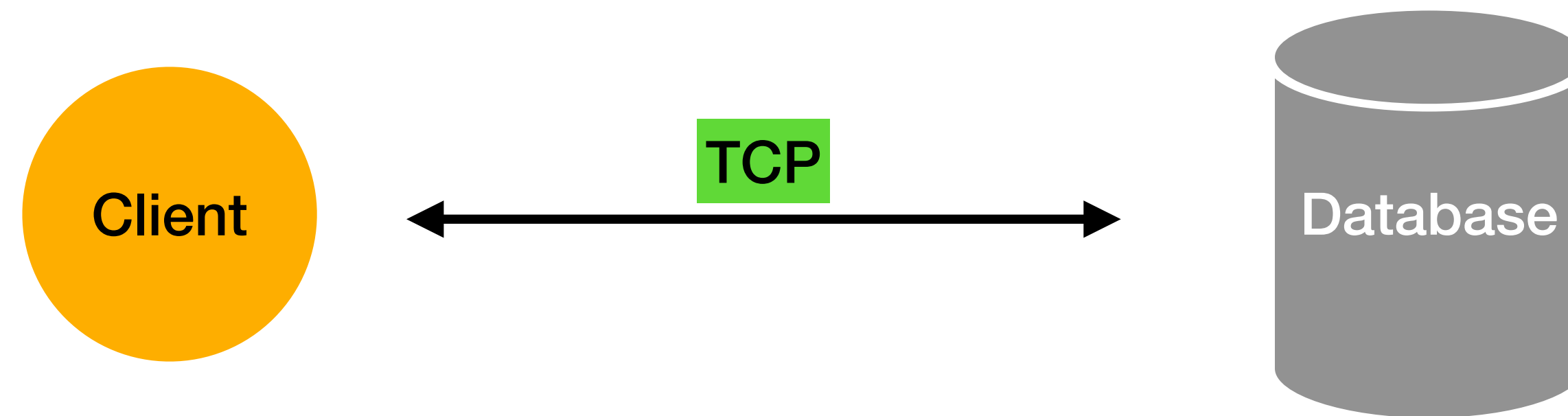
rest

tcp

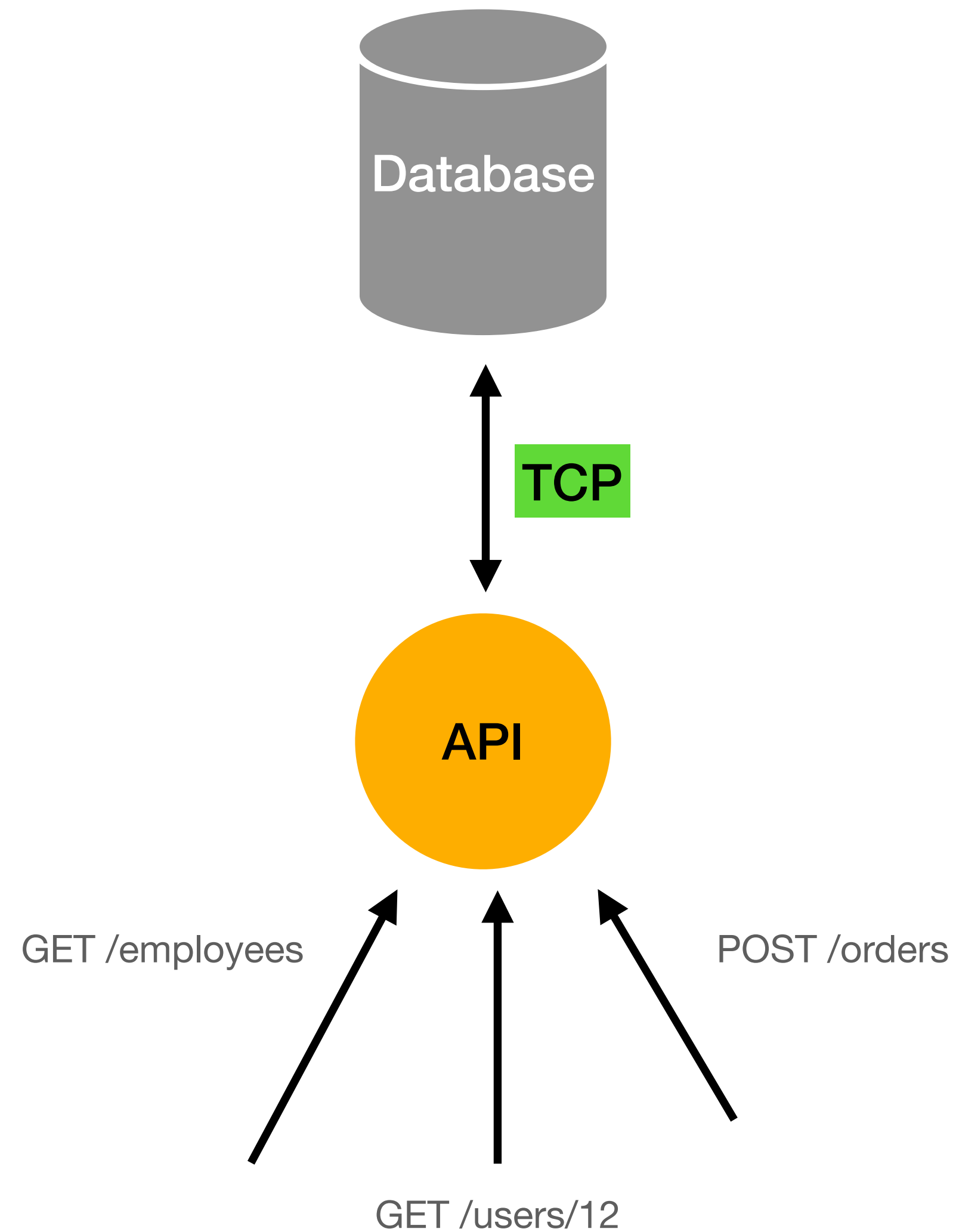
pooling

**Author: Héctor Valls**  
**<https://www.hvalls.dev>**

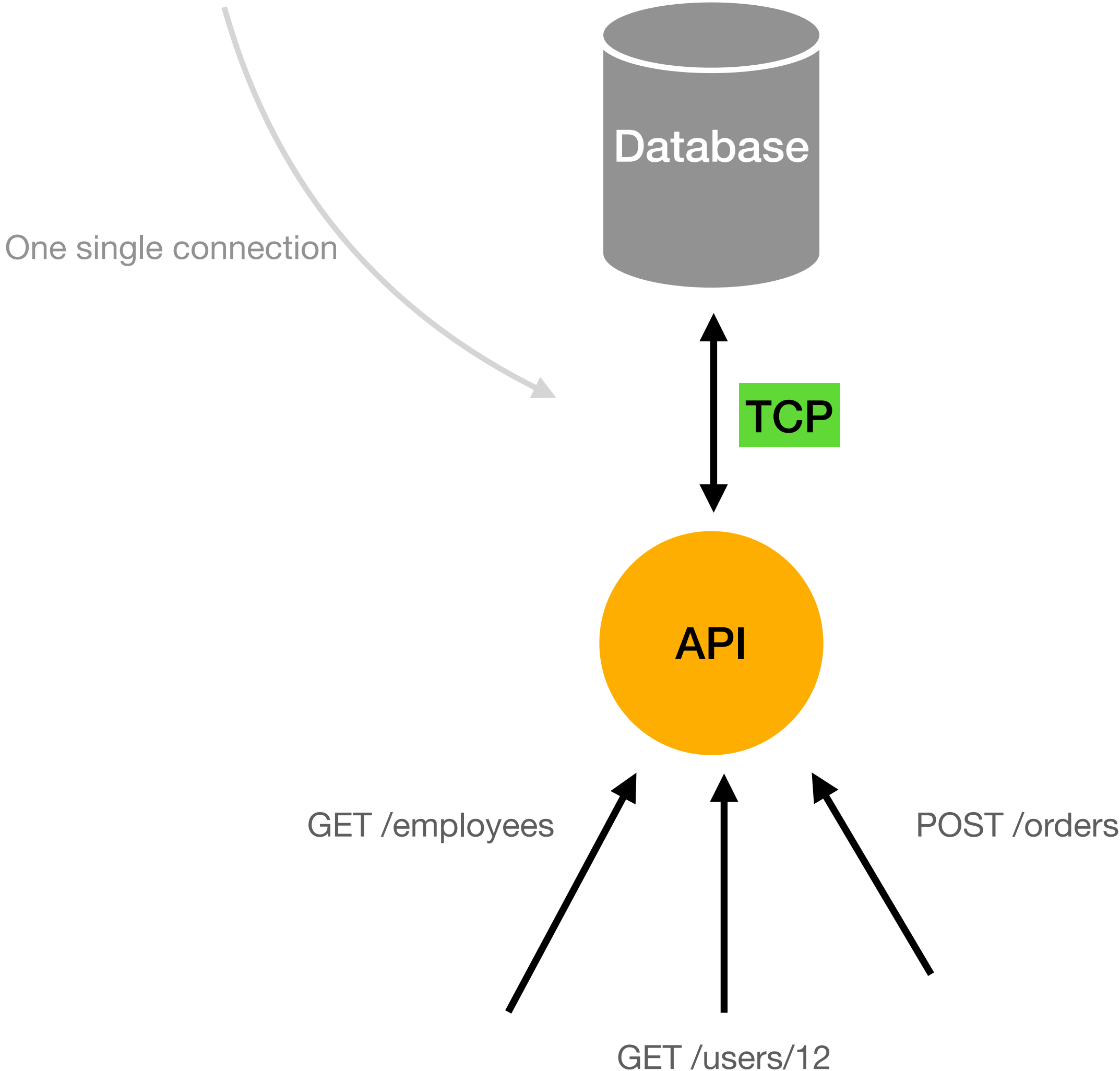
**Most of the database management systems uses TCP as the client/server communication protocol**



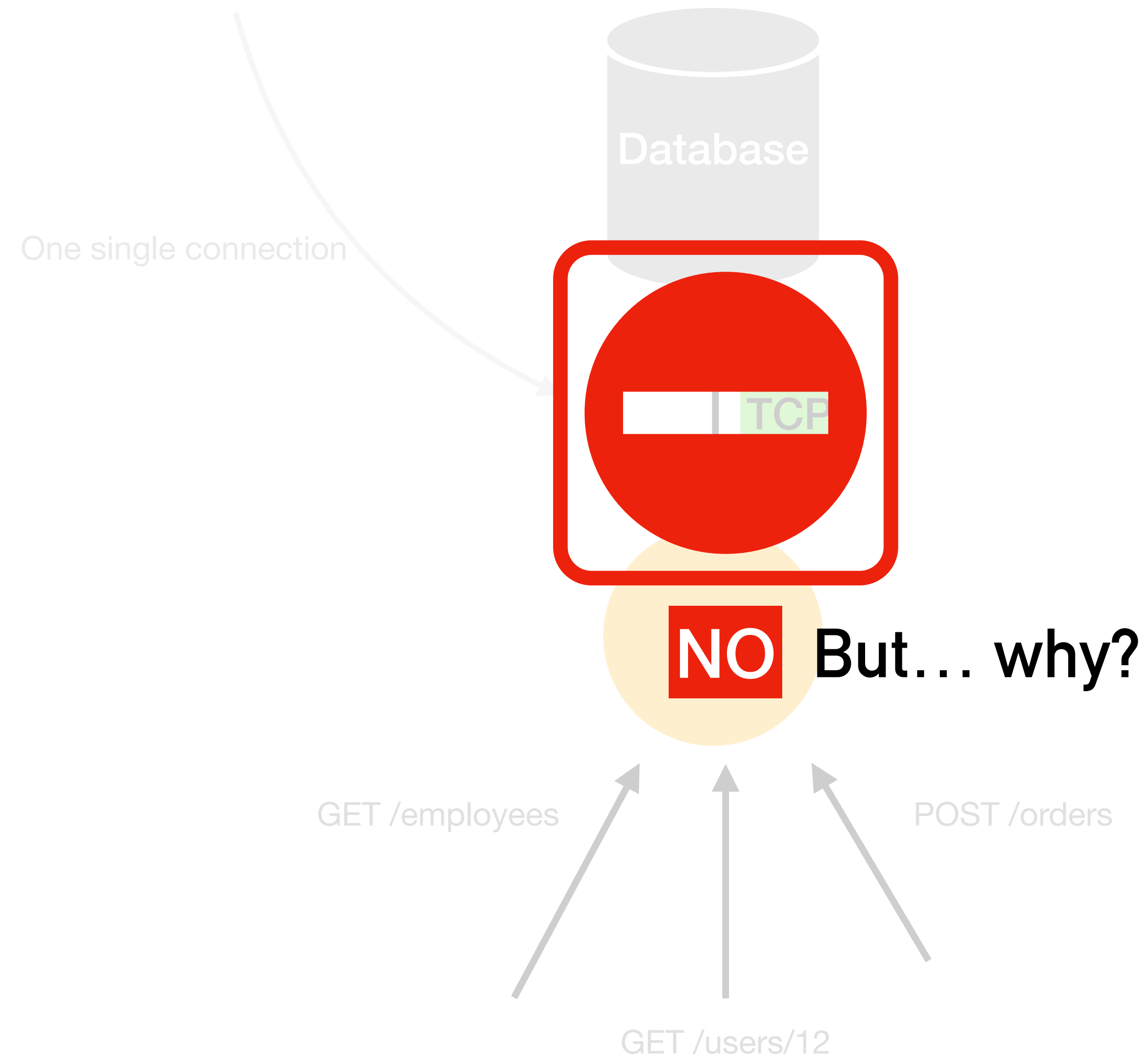
Say we are developing a REST API that will accept multiple HTTP requests and will query the database



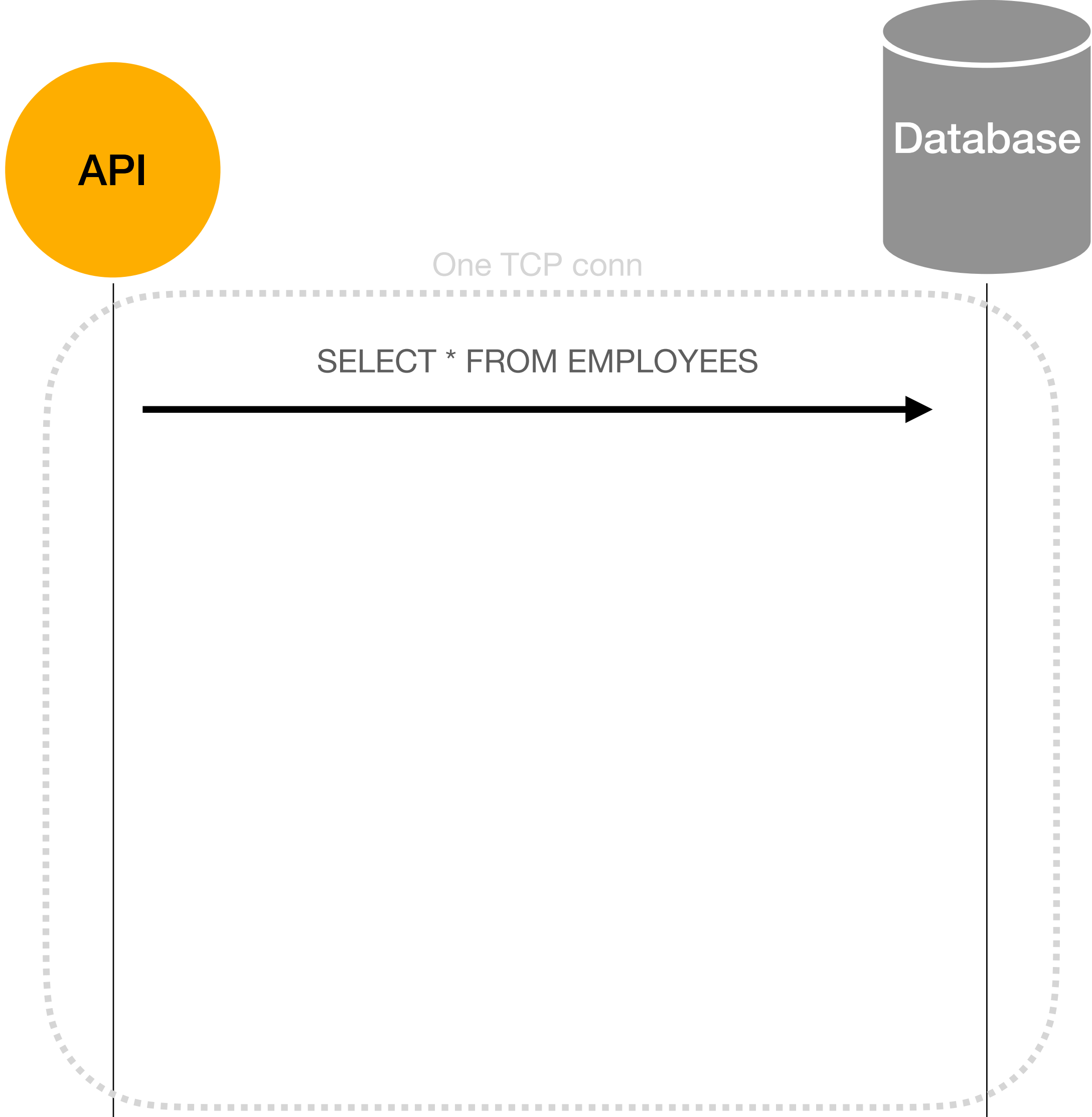
# Can we use one single TCP connection to database to handle every query?



# Can we use one single TCP connection to database to handle every query?

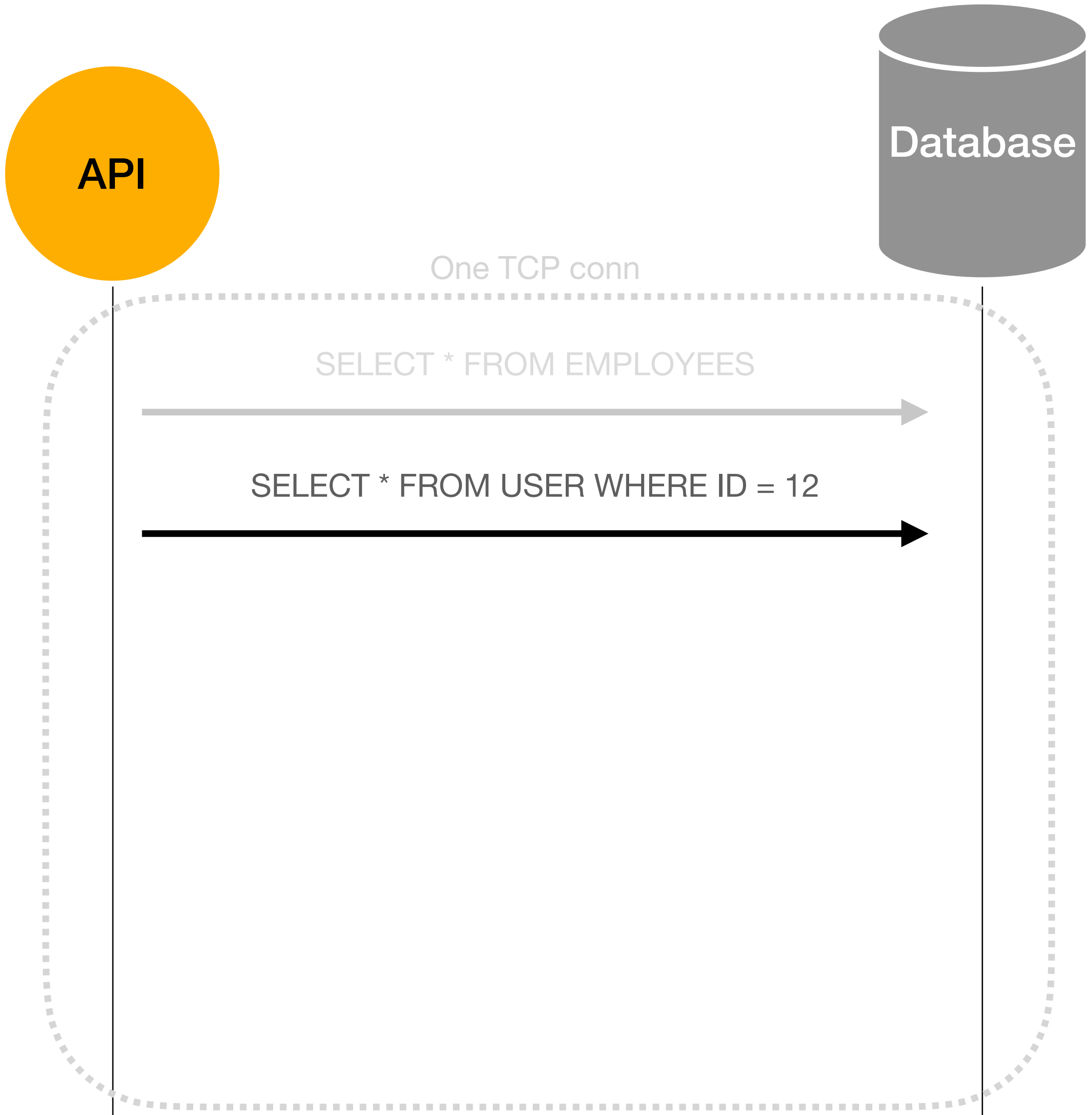


Imagine the API makes a query to fetch employees



Imagine the API makes a query to fetch employees

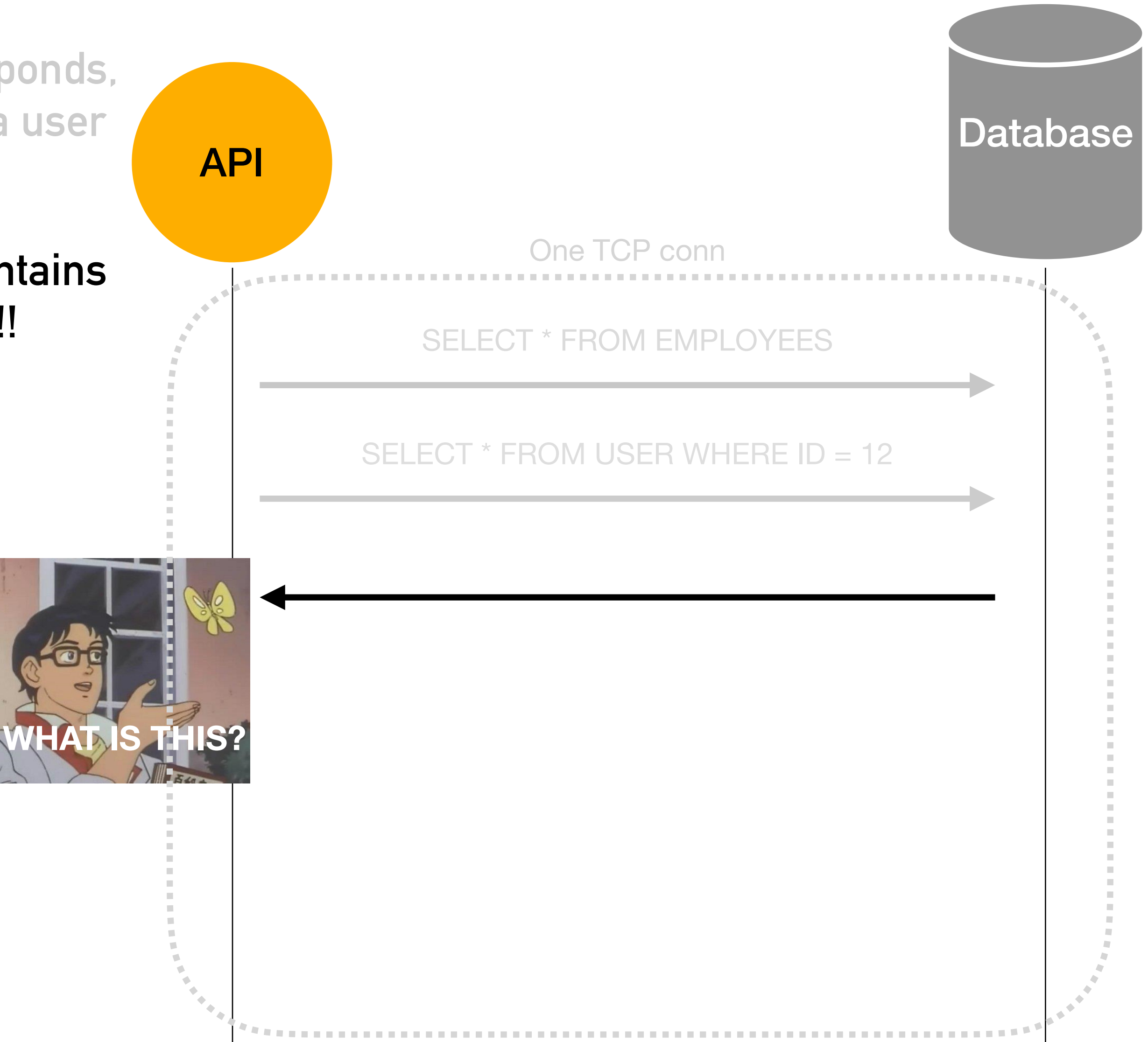
And, before database server responds, another query is made to fetch a user



Imagine the API makes a query to fetch employees

And, before database server responds, another query is made to fetch a user

**When server responds, API doesnt know if response contains the employees or the user!!**

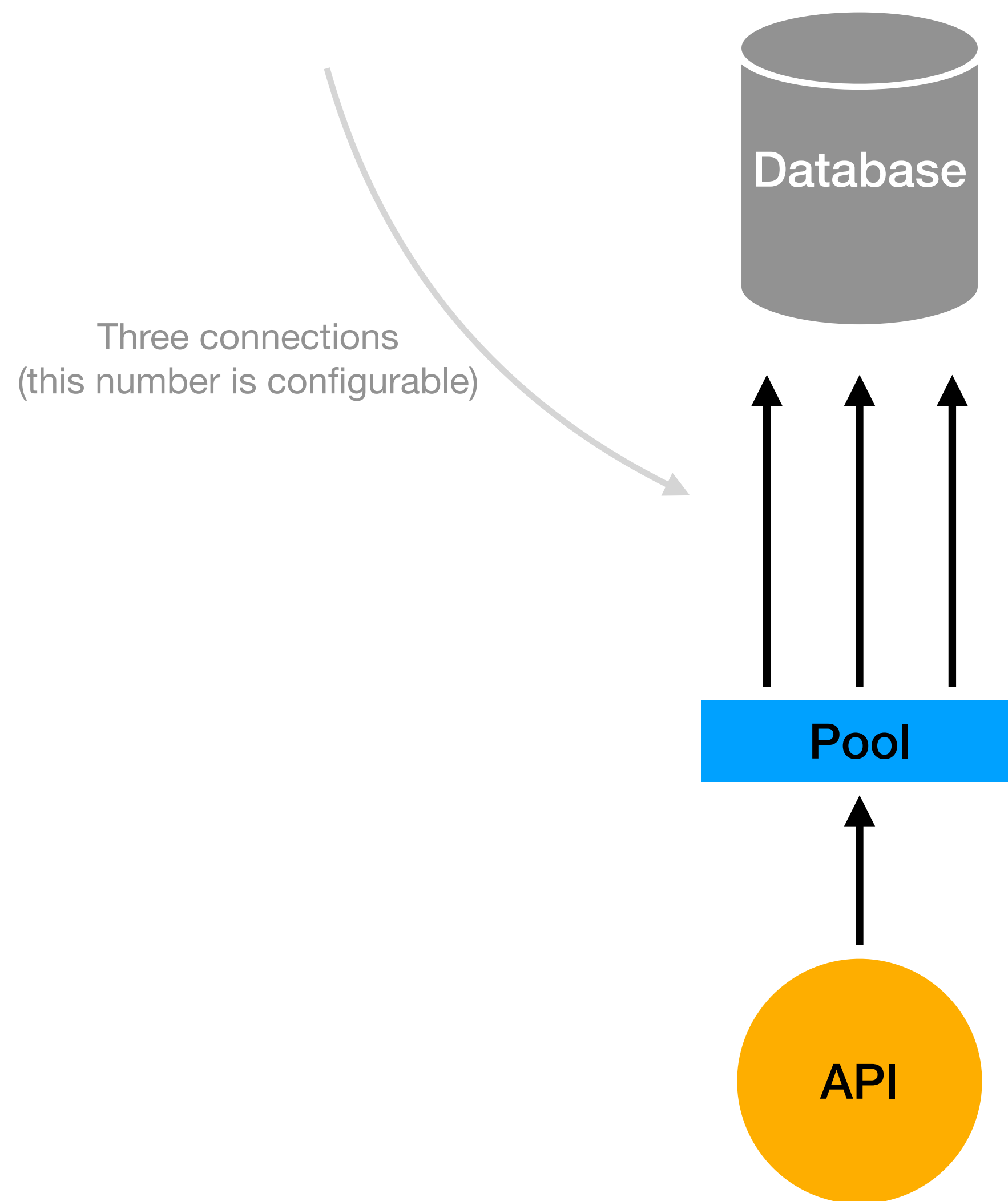




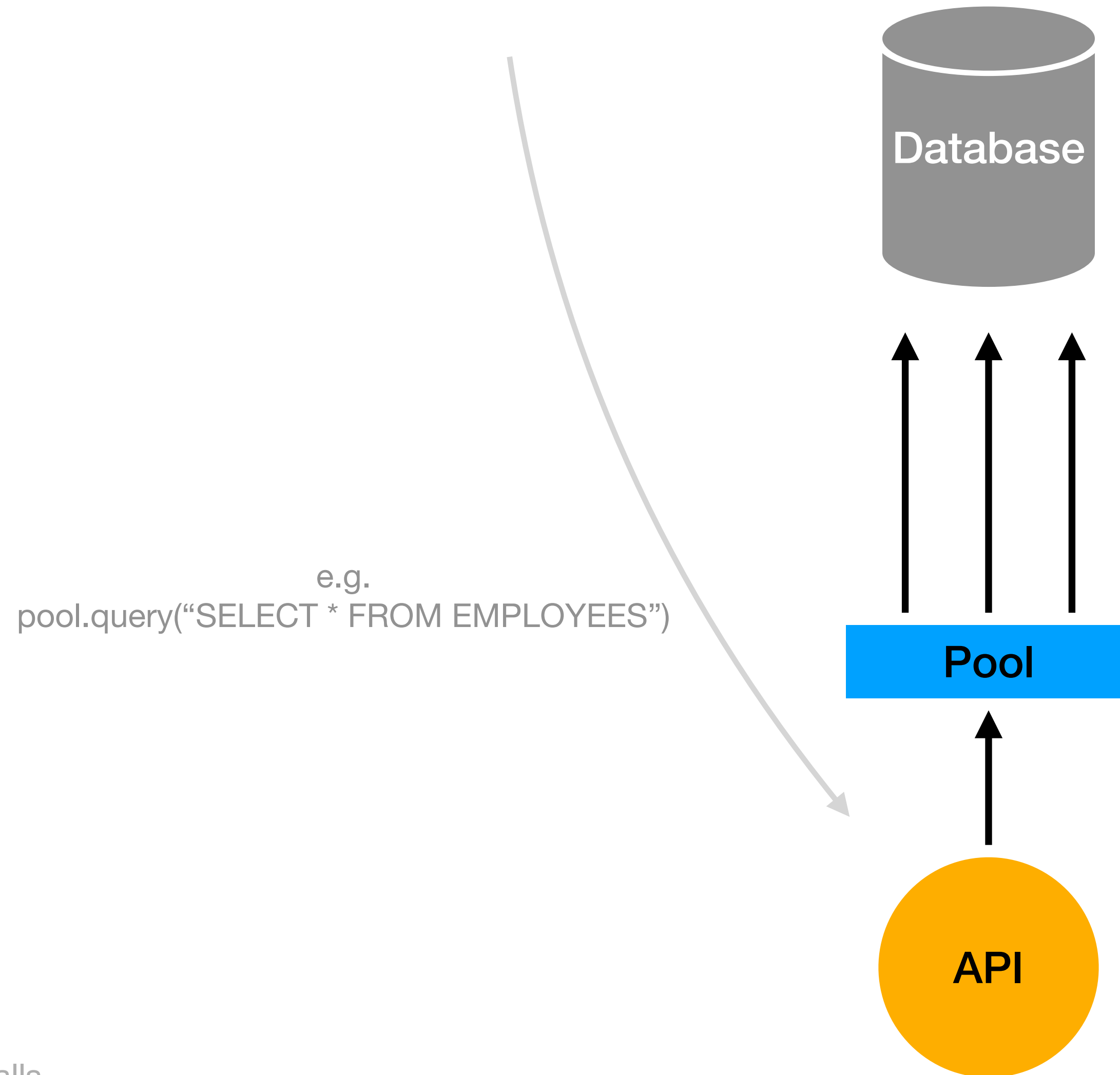
How can we solve this?

Using a connection pool

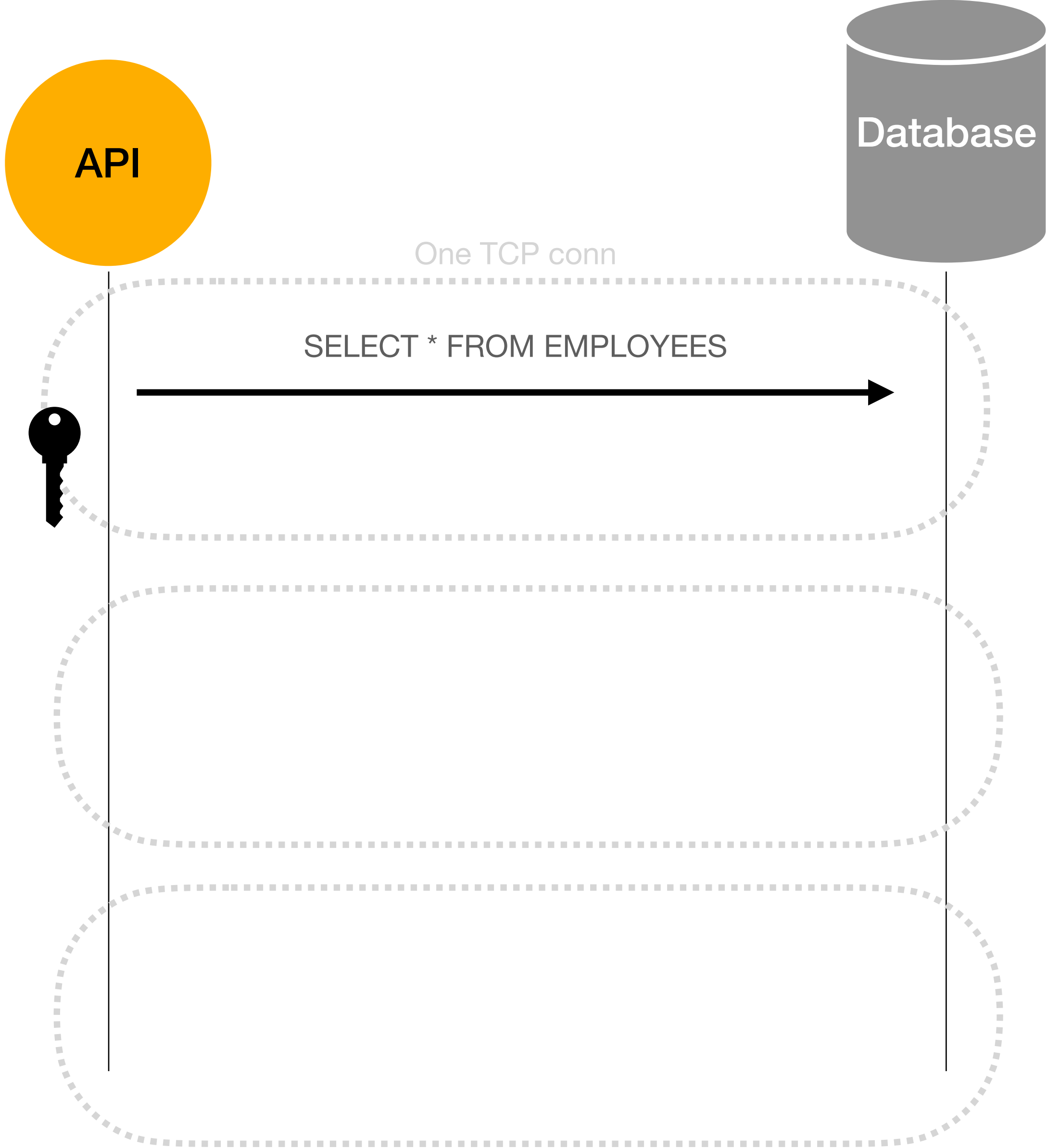
# A connection pool opens several connections to database and manage them smartly



**Database client (API) knows nothing about connection management.  
It just interacts with the connection pool**

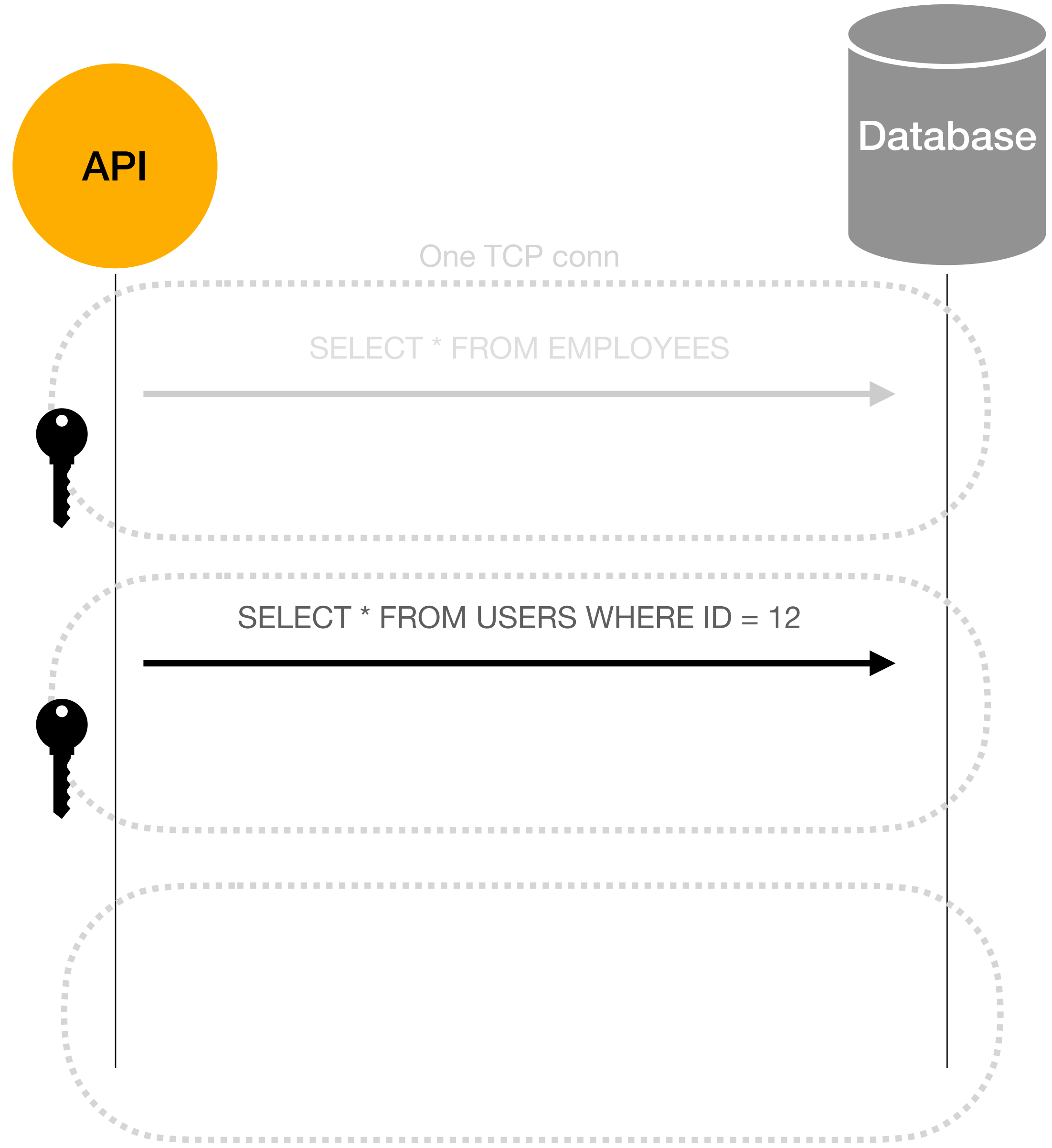


When API makes a query to fetch employees, pool reserves a connection



When API makes a query to fetch employees, pool reserves a connection

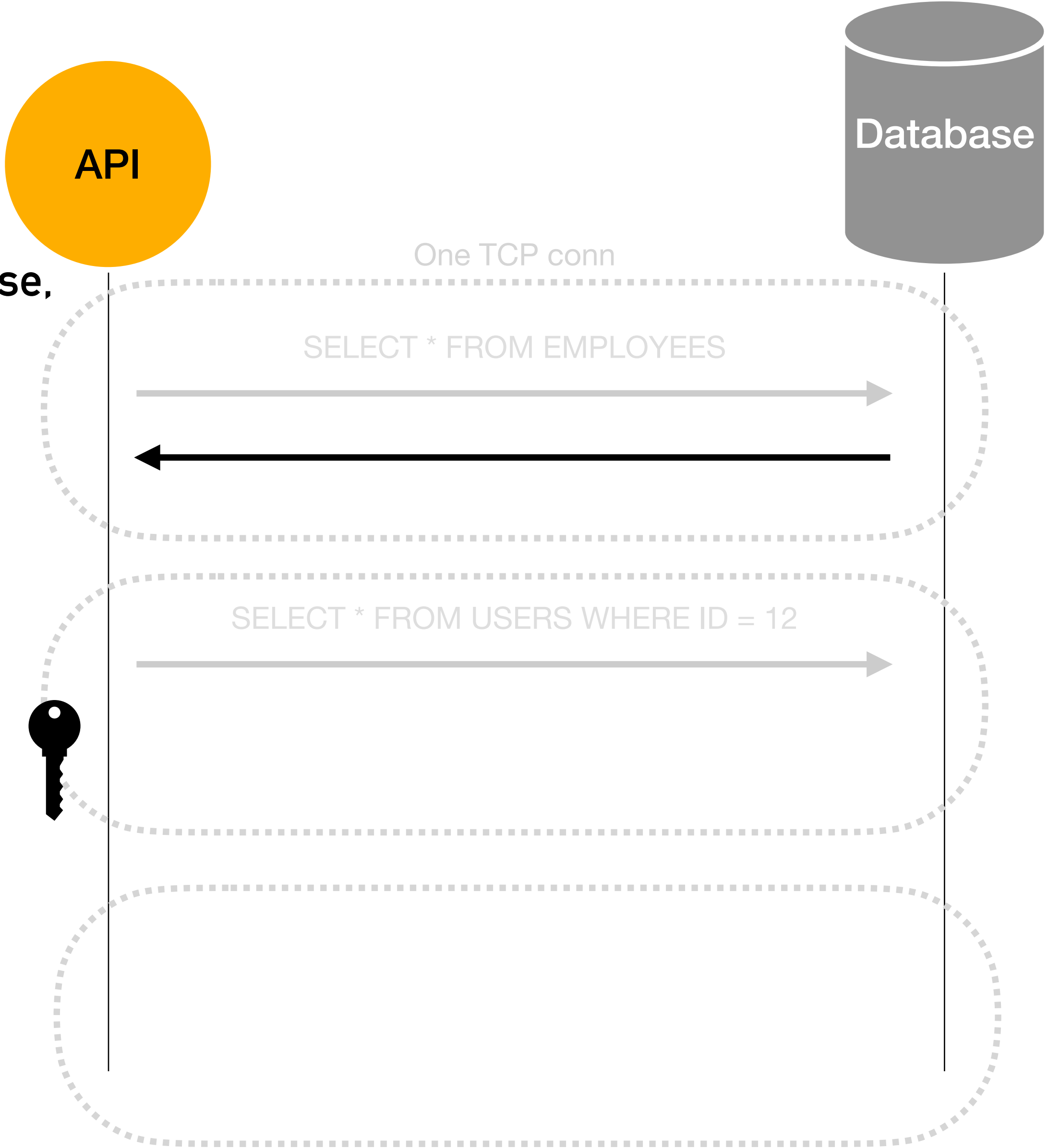
If, before database server responds, another query is made to fetch a user, pool will use another connection



When API makes a query to fetch employees, pool reserves a connection

If, before database server responds, another query is made to fetch a user, pool will use another connection

When first query gets the server response, pool will release the connection



**Every serious database client offers a way to create a connection pool.**

**Delegate to them connection management and avoid communication issues.**

**Find more content like this on  
<https://www.hvalls.dev>**