COMP1531

🐶 Software Engineering

5.3 - Development - Persistence

In this lecture

Why?

- Data is a critical part of nearly every software system, so let's understand it's interactions

What?

- Data
- Persistence
- Persistence in python

Data

Data: Facts that can be recorded and have implicit meaning

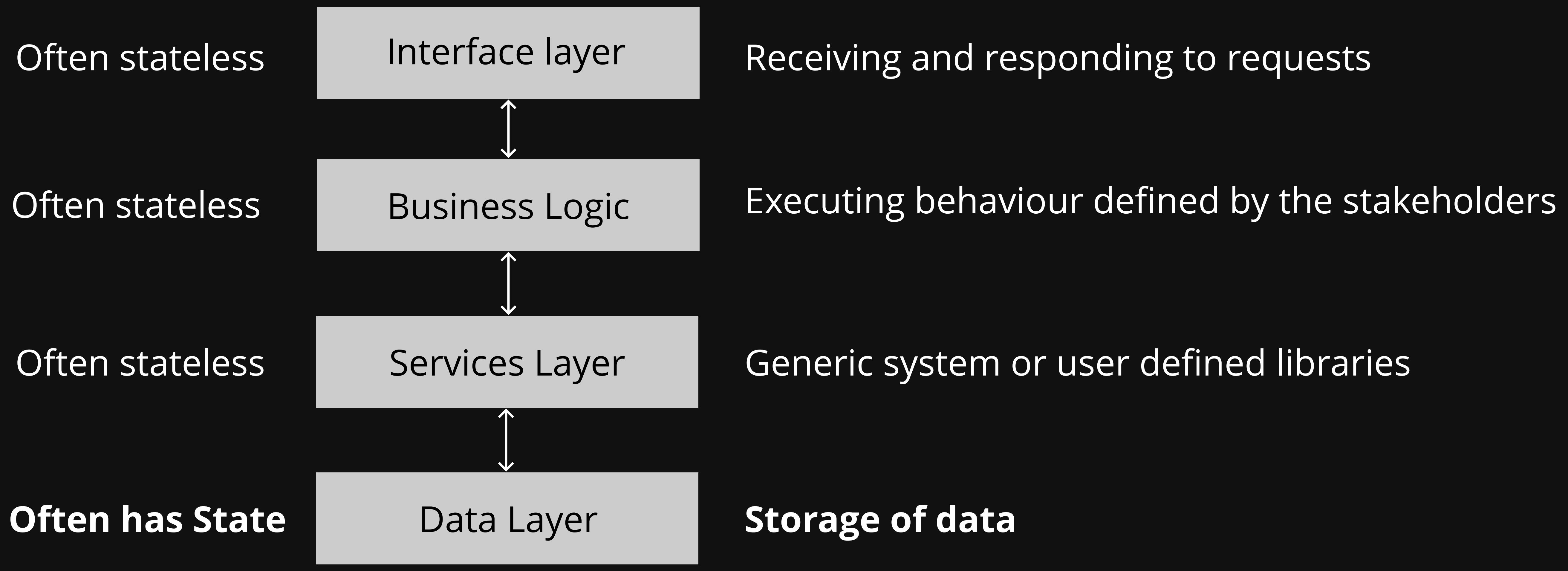
From **data (raw)** was can find **insights (information)** that allow us to make **decisions**.

Data (and "big data") are becoming huge topics in the world of computing and mathematics. This course does not address these topics.

Data in Applications

Data is part of every part of software (e.g. variables).

However, often when we refer to "Data" in software we're referring to a "Data Layer" - a part of the software that is focused solely on housing data.



Often stateless	Interface layer	Receiving and responding to requests
Often stateless	Business Logic	Executing behaviour defined by the stakeholders
Often stateless	Services Layer	Generic system or user defined libraries
Often has State	Data Layer	**Storage of data**

Data Layers ("Databases")

Database: A word used to describe a store of data. Databases are typically what a data layer consists of.

There are 3 main ways to store data:

1. In-memory (non-persistent)
2. In-file (filesystem direct storage)
3. In-database (Relational SQL, NoSQL)

As you move down the list, barrier to entry becomes higher, but so does performance.

To learn more about relational SQL, study COMP3311.

Storing Data: Persistence

Persistence: When program state outlives the process that created it. This is achieved by storing the state as data in computer data storage.

Persistence can be interpreted very broadly, though in this course we will define it as storing data to disk.

Storing Data: Persistence

Can we modify our project server to persistently store data? How would we do that?

Python specific - pickling

A very common and popular method of storing data in python is to "pickle" the file.

- Pickling a file is a lot like creating a .zip file for a variable.
- This "variable" often consists of many nested data structures (a lot like your iteration 2 data)

Let's look at an example

pickle_it.py

unpickle_it.py

Feedback