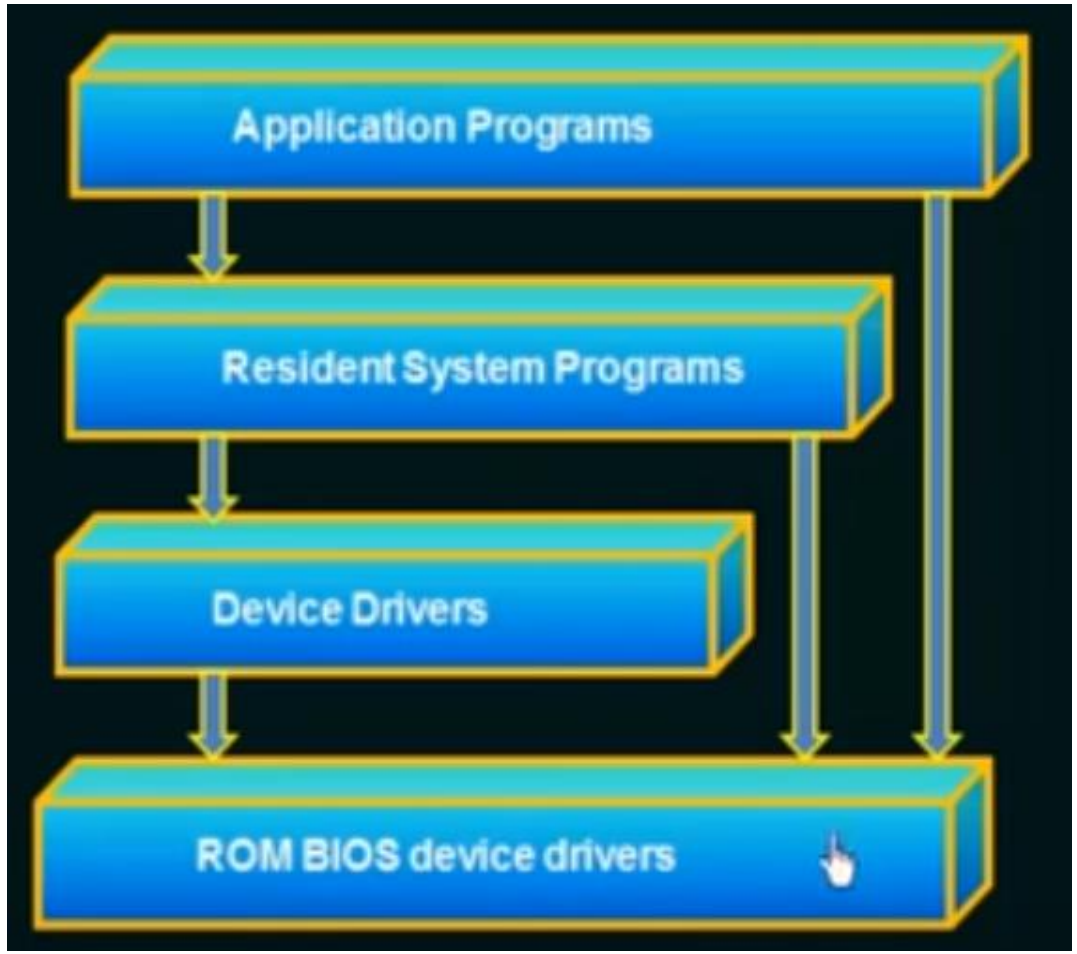
Structures of OS

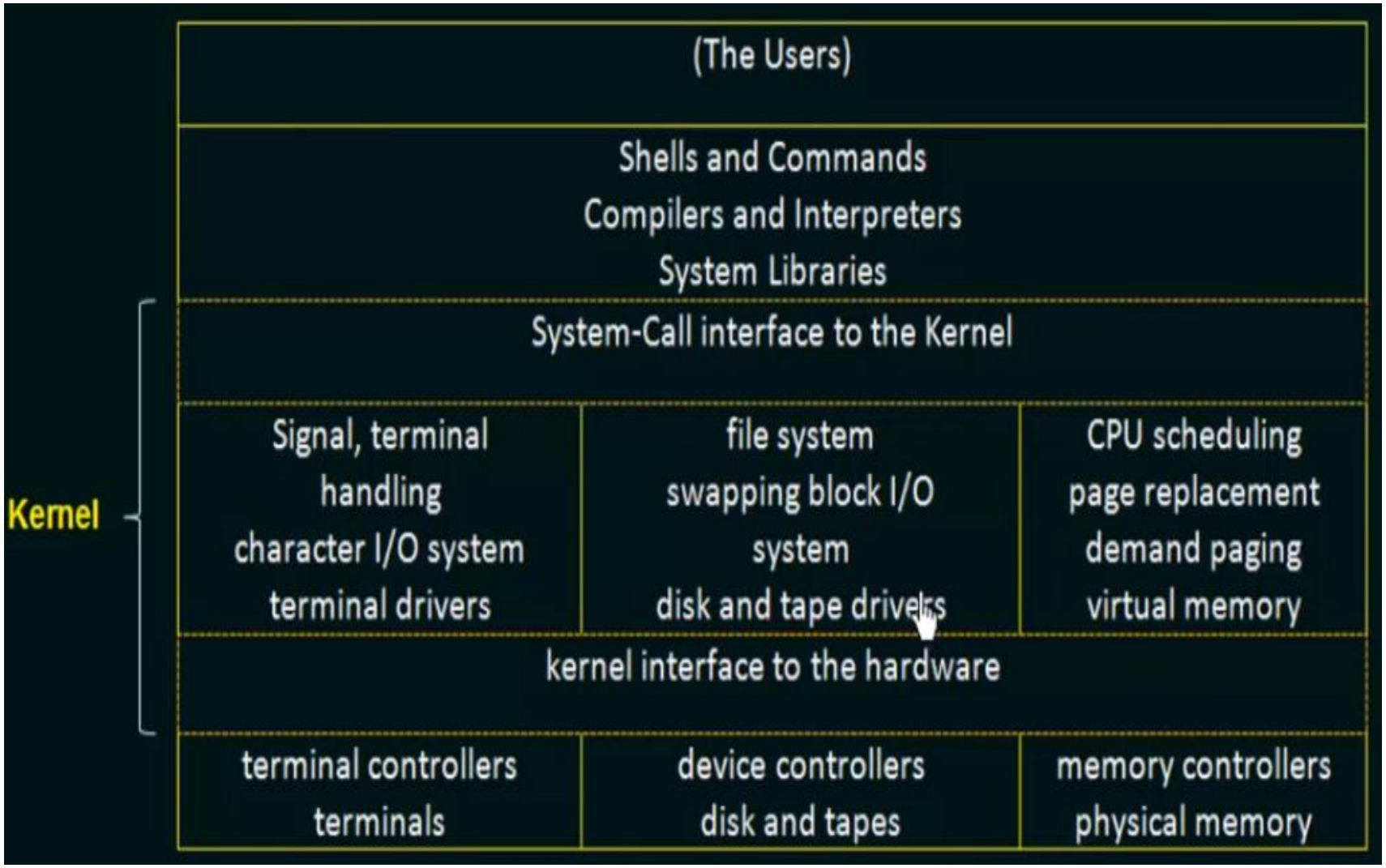
As OS has a large and complex structure, hence it has to be engineered very carefully so that it can function properly and can be modified easily.

Simple Structure



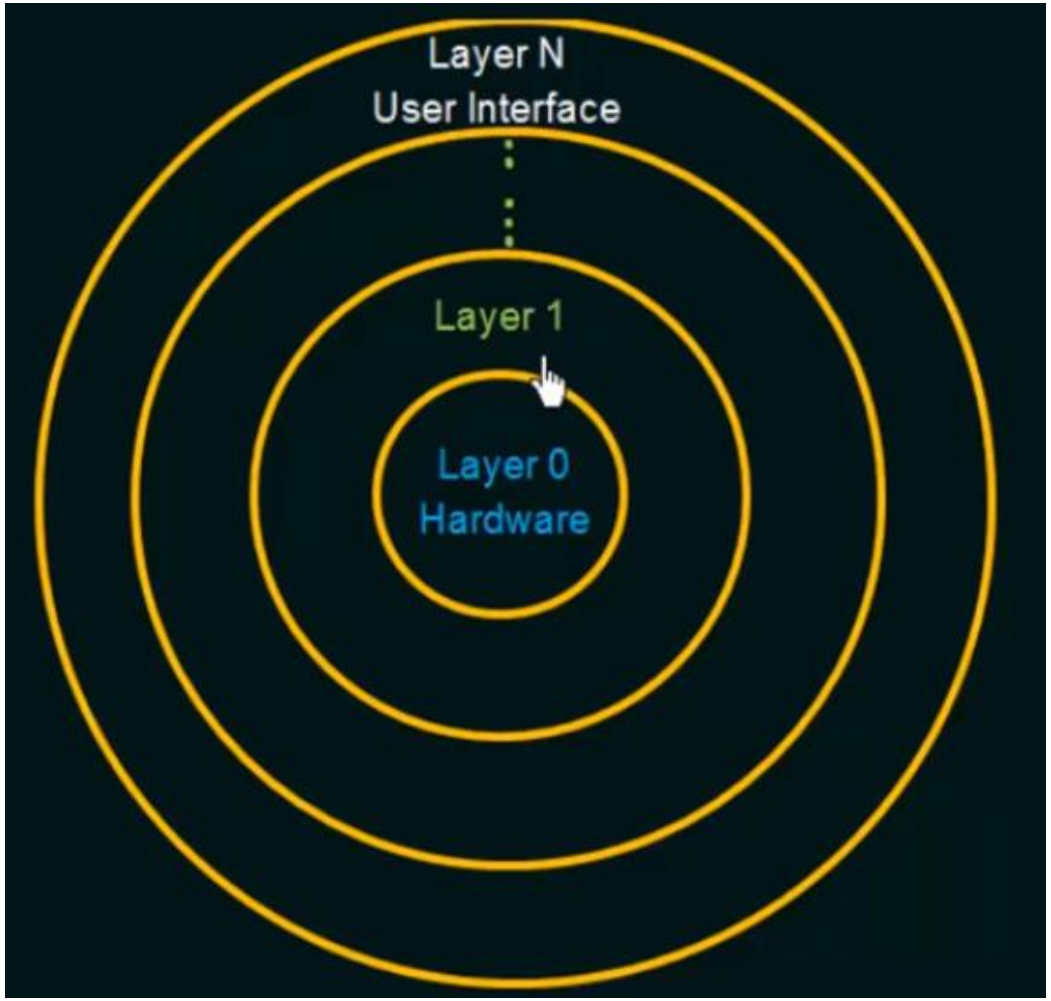
- It was used in very beginning and old OS
- Functionality of interfaces and levels are not well separated
- System crash, vulnerability
- Not well protected, not well structured
- Written on Intel 8088
- It is followed by MS DOS system

Monolithic Structure



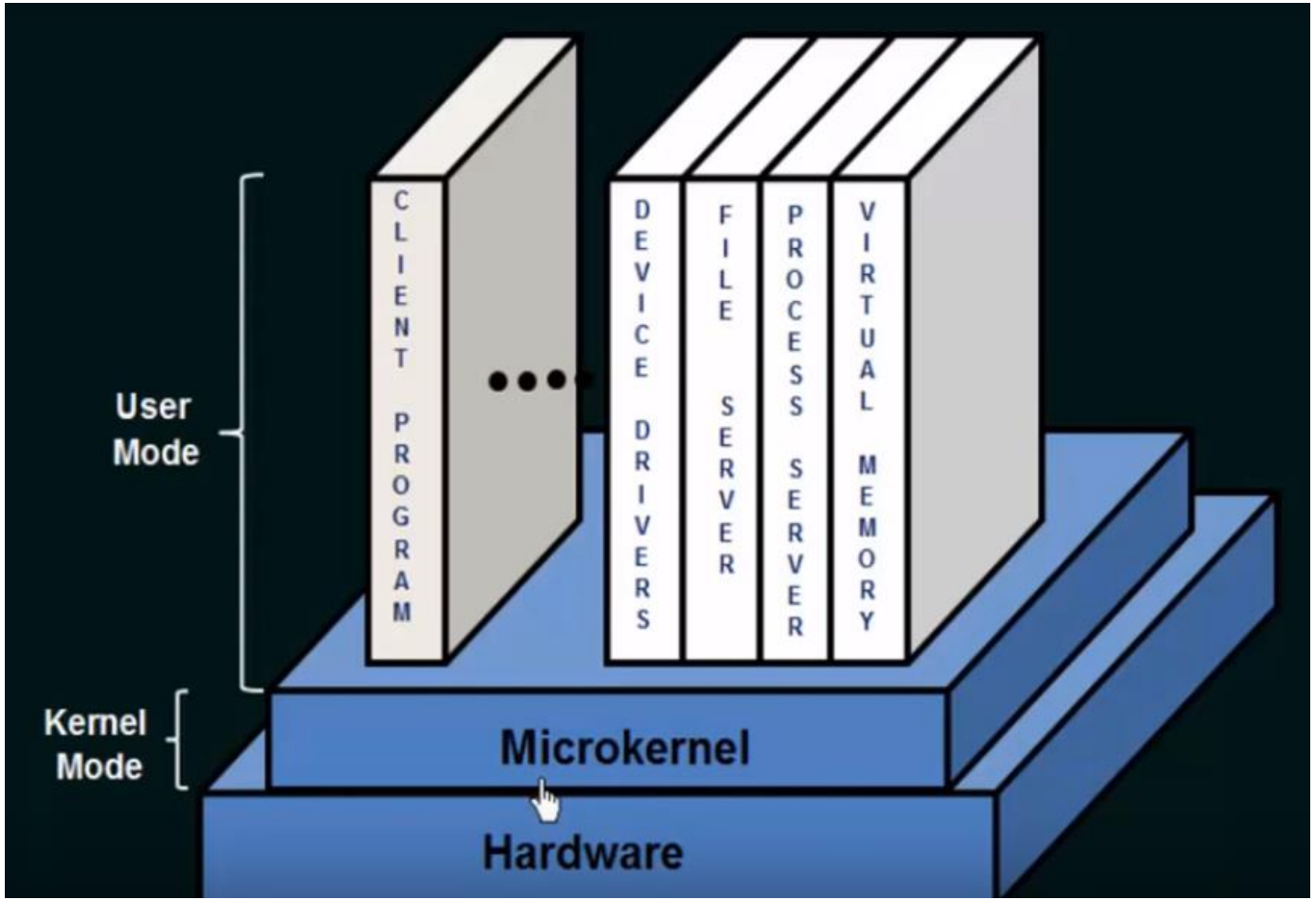
- It is followed by earlier Unix system
- Implementation, Modification and Debugging is difficult because all functionalities are packed in one level

Layered Structure



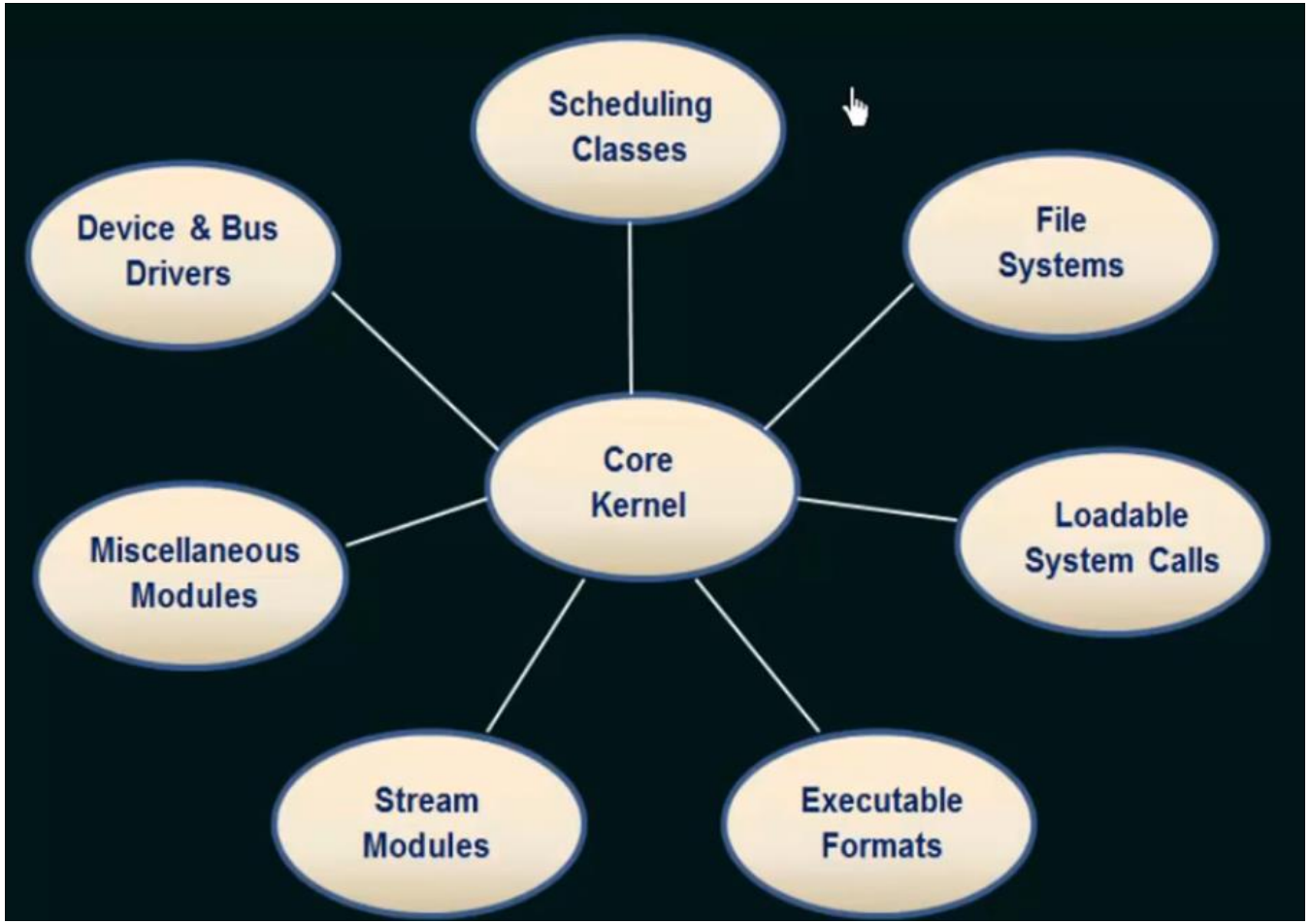
- Structure is divided into number of layers
- As compared to monolithic kernel, in the layered structure the functionalities are broken down into different layers
- Easy to implement and maintain
- Designing and deciding which of the layer should be top of/bottom of any particular layer
- Any Layer can use the services of its underneath layers.
- Not efficient/ services may be late
- Hardware is protected from the layer above.

Microkernel Structure



- Remove the non essential component from the kernel and implement them as user level or system level programme.
- Role of the microkernel provide the communication between different system/user programmes through message passing.
- Crashing problem of an entire system is not going to happen
- Suffer from performance decrease due to increase system overhead

Module Structure



- Functionalities are available as modules which are loaded into kernel either at boot time or run time. (dynamically loaded as and when required)
- Protected interface
- Flexible as compared to layered approach because any module can call any other module
- No need of message passing hence less system overhead.