Logical Architecture

Layered Architecture

? Architecture in UP

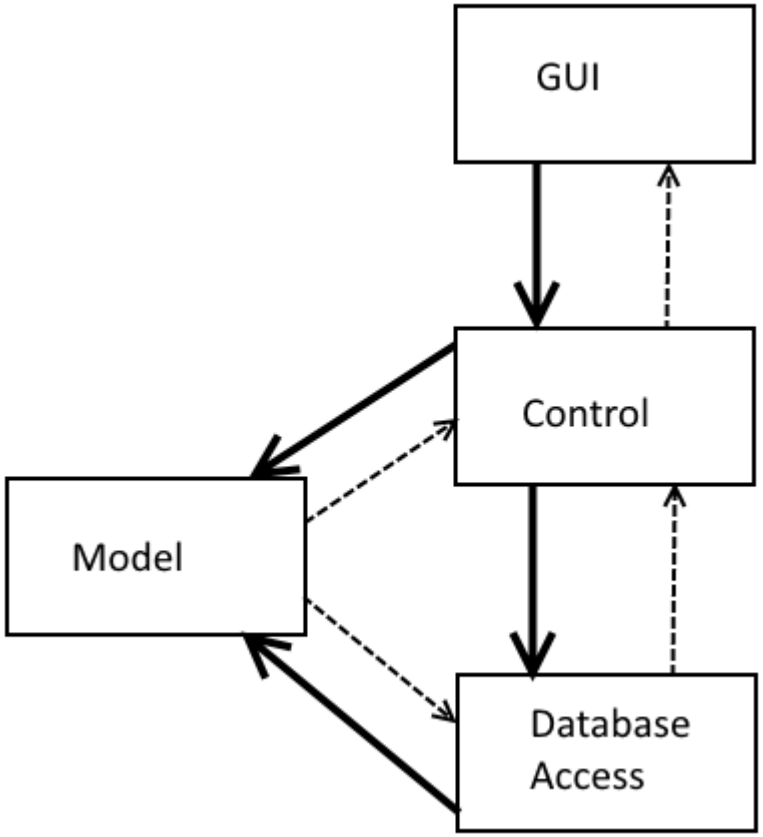
- In short:
  - Overall design decisions
  - Software elements and how they are related
- Considerations start up in inception
  - Candidate architecture
- Established in elaboration
  - Layering, patterns etc.
  - Verified through design, implementation and test of architecturally significant use cases (the most complex)

Logical architecture and layers

- Logical architecture
  - Large-scale organization of the software classes into packages/namespaces, subsystems and layers
- Layer
  - Very coarse-grained grouping of classes, packages, or subsystems that has cohesive responsibility for a major aspect of the system
  - Organized such that higher layers call upon services of lower layers, but not vice versa
    (strictly layered <-> relaxed)

[Larman] chapter 13

Layered architecture (DAO)



GUI layer handles interaction between actor and GUI

Control layer handles execution of a use case

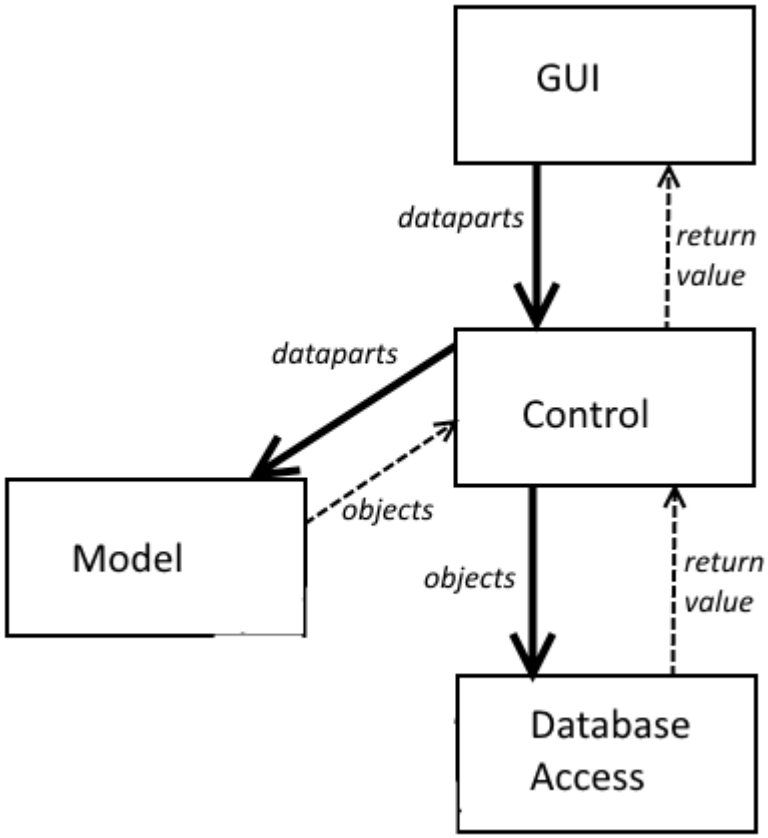
Model layer handles the domain classes which are used by the use cases

Database Access layer incapsulates the persistence (sql) of the domain classes.

In a closed architecture the DB layer would be placed underneath the Model layer

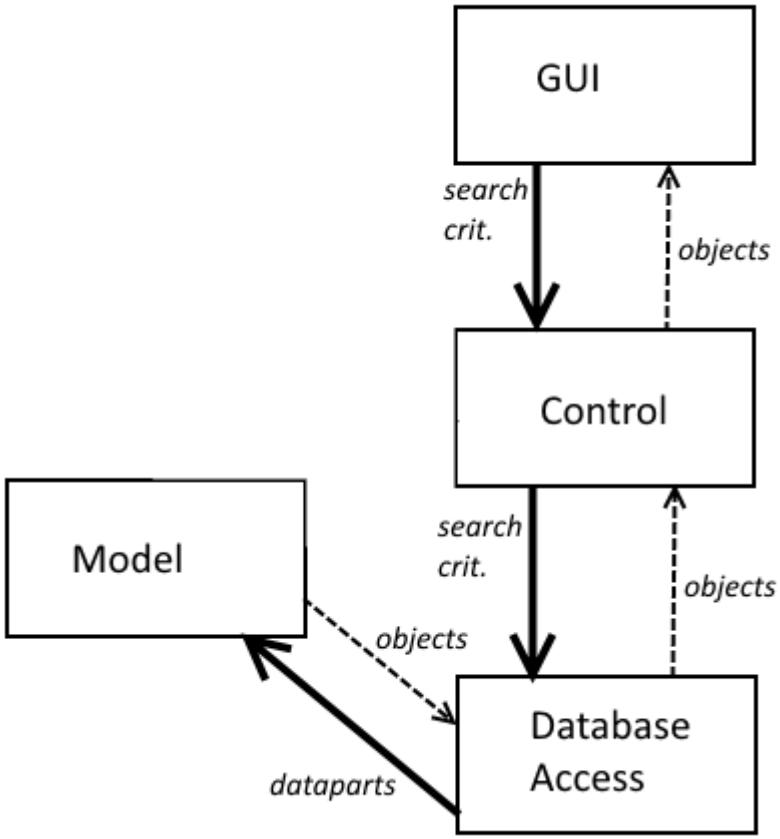
Layered architecture (DAO)

Save

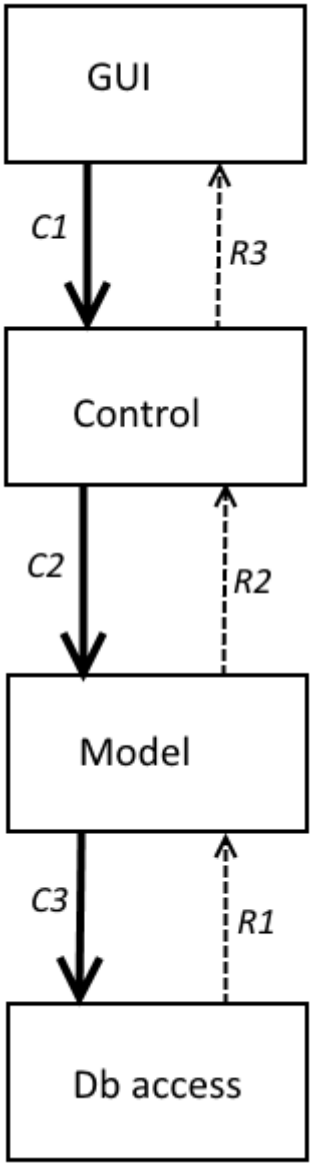


Layered architecture (DAO)

Read



Strictly layered architecture



Layered architecture

Consider the Layered architecture (DAO)

Explain how a save is performed?

Explain how a read is performed?

Consider Strictly layered architecture

Read: which kind of data is C1, C2, C3 and R1, R2, R3?

Save: which kind of data is C1, C2, C3 and R1, R2, R3?