**S O L I D**

Single Responsibility

Open-Close:

We worry about expectations of clients from the exposed abstraction(s). We try to be extensible to meet extra expectations of client(s) but we don’t modify the existing code because we might break some expectations.

Liskov:

We worry about run-times. We again worry about clients’ expectations from the exposed abstraction(s). In a hierarchical structure all derived types should be able to be used as base type without breaking the expectations. We control this principle in 5 areas:

. Pre-conditions: cannot be strengthened

. Post-conditions: cannot we weakened

. Invariants: must be preserved

. Exceptions: no new one should be thrown in derived types

. Contravariance of method arguments and Covariance of return types

All pre/post conditions are mech sometimes to protect invariants. All of them are categorized in Business rules. Therea are mainly 2 ways of handling business rules:

. Structural: one-2-one or one-2-many relationships are good examples

. Behavioral: all pre/post conditions

Interface segregation

Dependency inversion