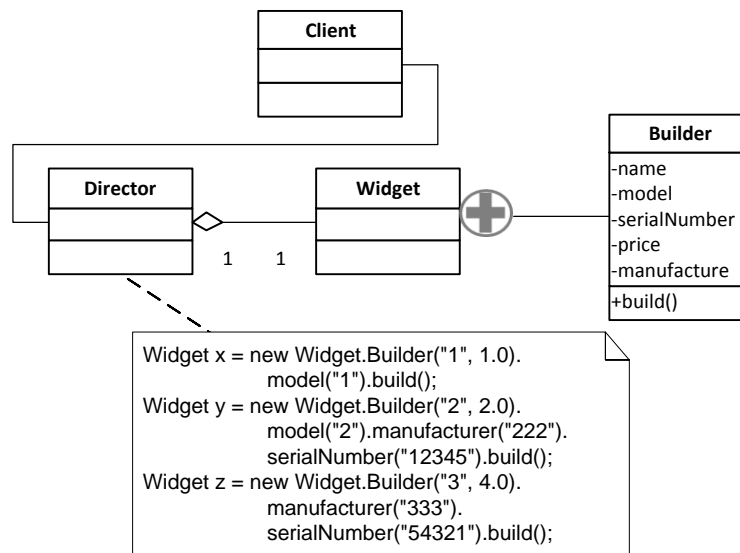


4 Builder Pattern [Gamma et al]

Builder pattern involves an instruction process that follows certain rules on how to create complex object. Usually applications contain complex objects that are created by part by part, each part is a separate entity and together they form a complete object . Builder pattern slice the operations of creating these parts in such a way defined by the buisness. It is basically a process that starts with the creation and ends with the final product. [1] defines the intent of builder pattern as “separate the construction of complex object from its representation”. [24] discusses an application of builder pattern. Following is the variant of builder pattern:

4.1 Instantiating builder object with multiple attributes - Nested Builder [81]

This variant of builder pattern provides an access to independent a client to create builder object of multiple constructor attributes that are optional. The variant is actually a solution to telescoping constructor and Java Beans pattern. The nested builder class holds the oppurtunity to get a builder object, the client sets parameters of its interest and make a find Build() call to generate the object.



4.1 UML Diagram generated from Source code at [81]