UML Class Diagrams

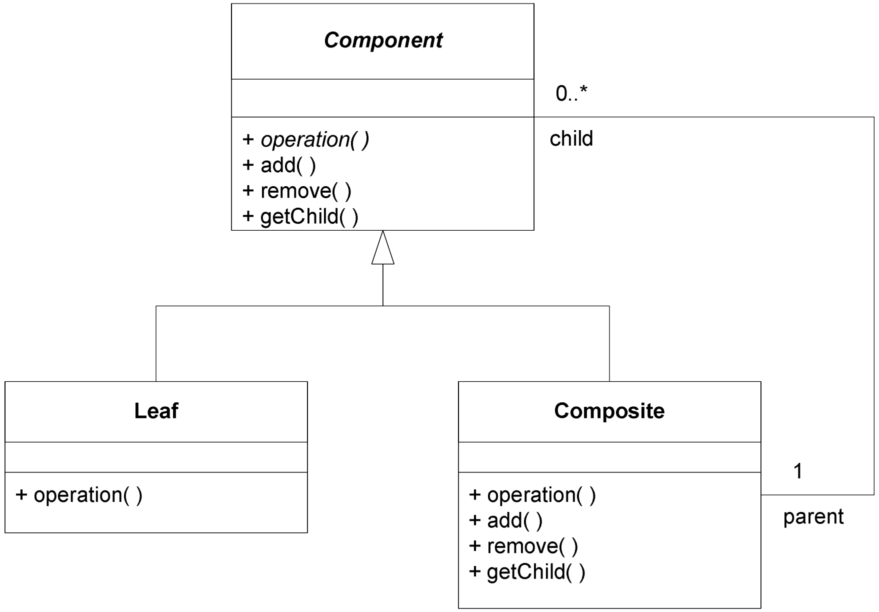
(Complements)

Design Patterns

- In software engineering, a **design pattern** is a general repeatable solution to a commonly occurring problem in software design. A design pattern is not a finished design that can be transformed directly into code.
- It is a description or template for how to solve a problem that can be used in many different situations.
- Object-oriented design patterns typically show relationships and interactions between classes or objects, without specifying the final application classes or objects that are involved.

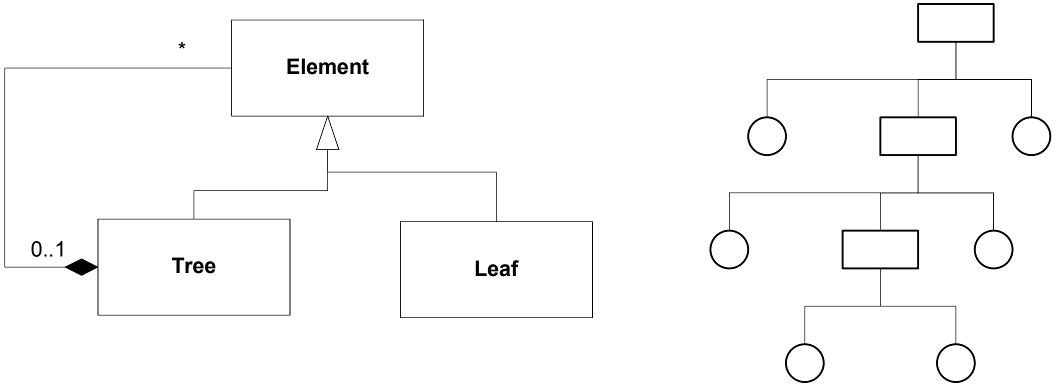
Composite Pattern

- In computer science, the **composite pattern** is a structural design pattern.
- Composite allows a group of objects to be treated in the same way as a single instance of an object.
- The intent of Composite is to "compose objects into tree structures to represent part-whole hierarchies. Composite lets clients treat individual objects and compositions uniformly."

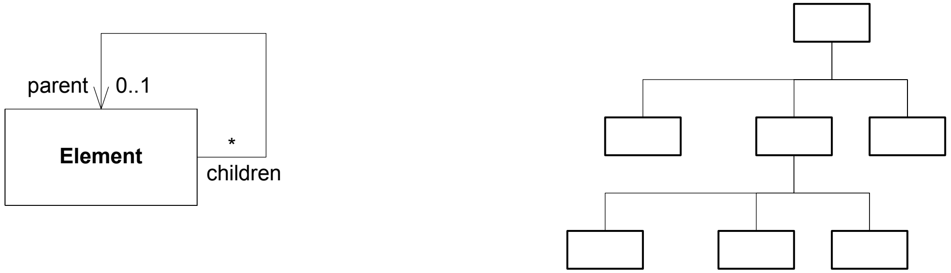


Trees & Graphs

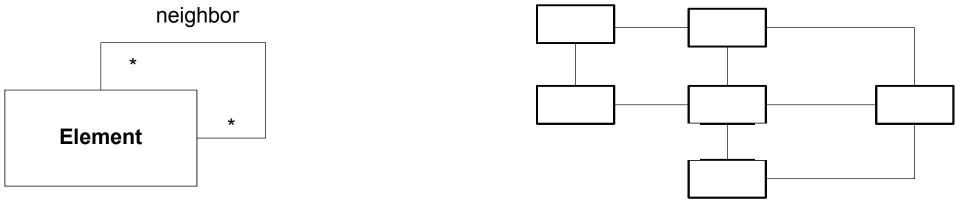
Differentiated Tree



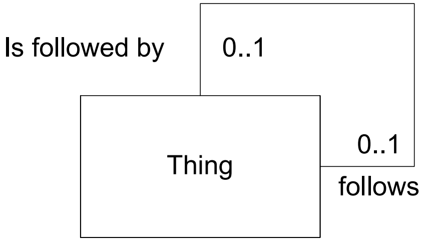
Undifferentiated Tree



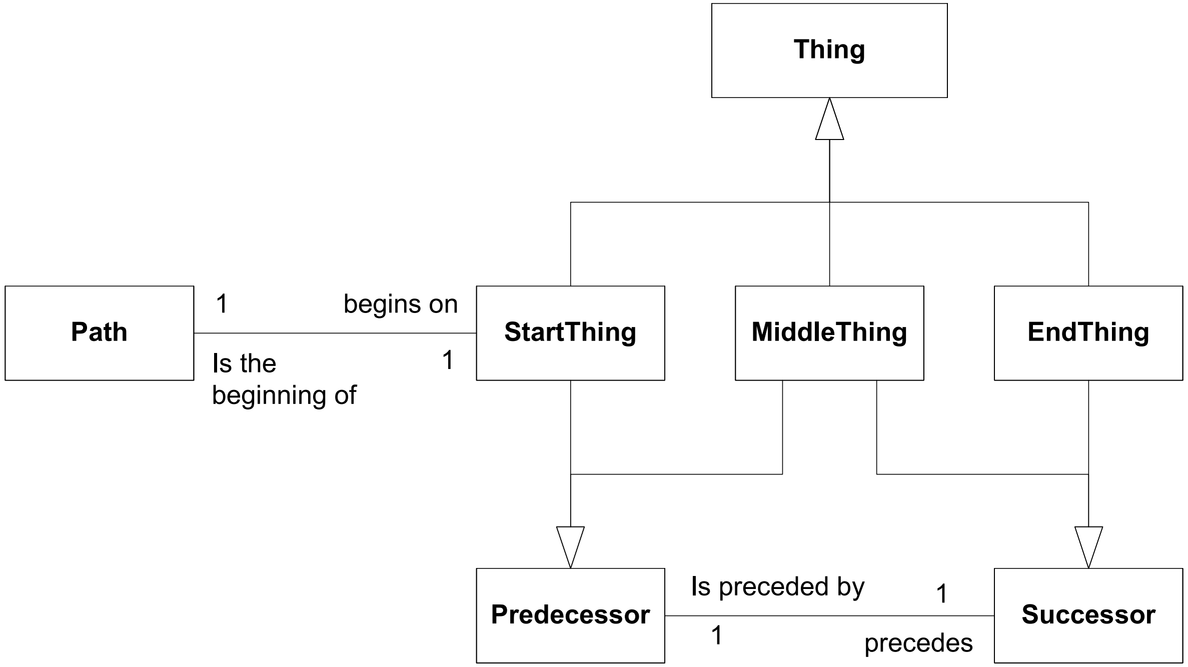
Graph



Linear Pattern



Structure is not enough.
OCL constraints needed


Thing
Path
1
begins on
Is the beginning of
1
StartThing
MiddleThing
EndThing
Predecessor
Is preceded by
1
1
precedes
Successor