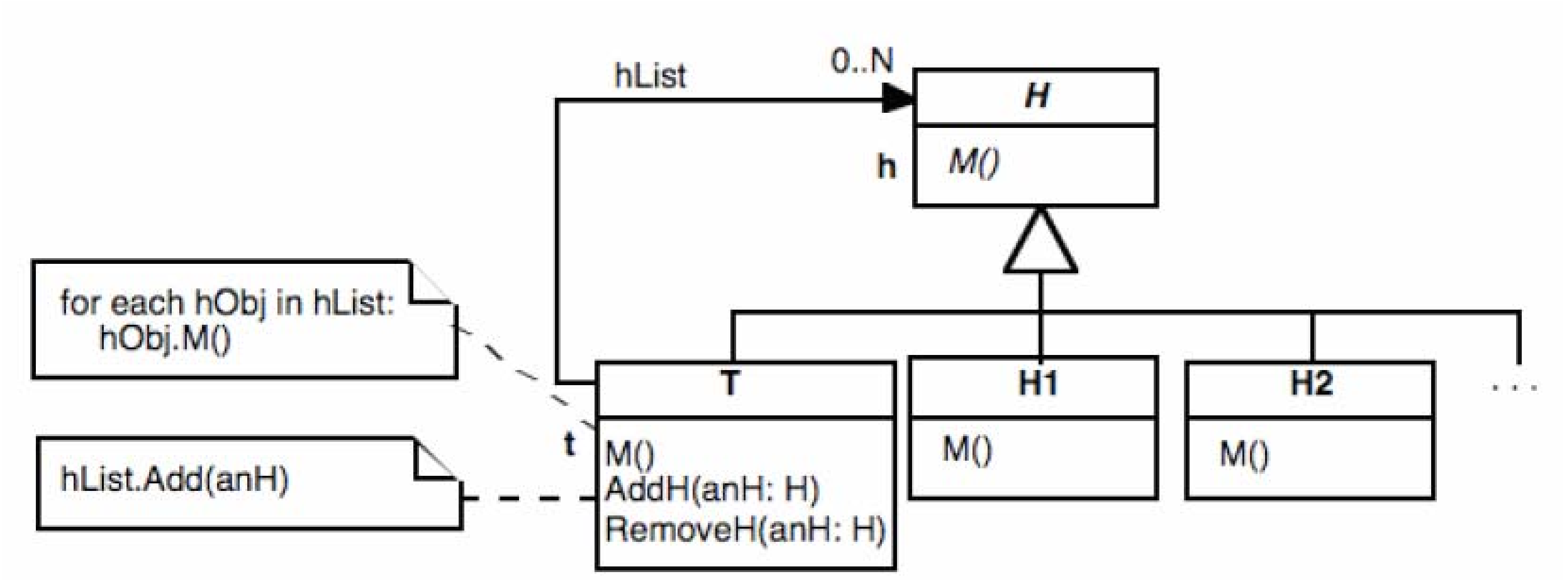
Construction of Flexible Software

- **Composition**
- Decorator

UNIVERSITÄT SALZBURG

The Composite Construction Principle

Composite: A tree of objects can be used like an individual object

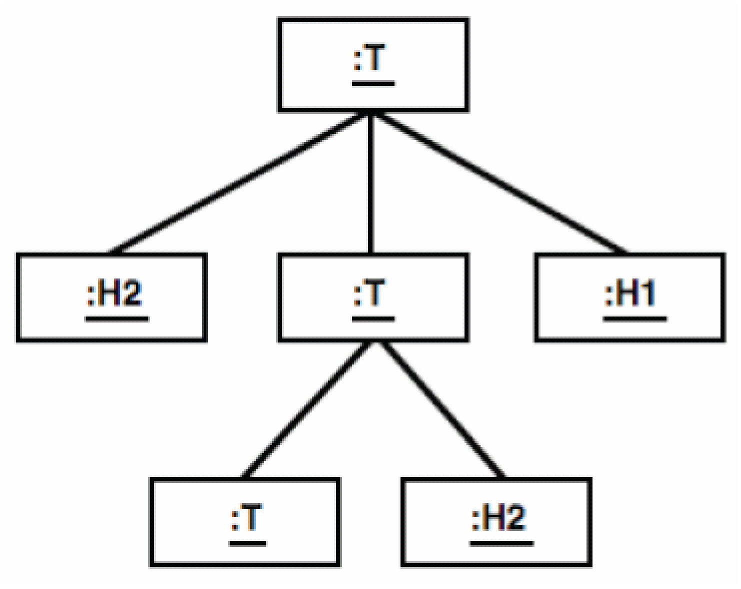


- The names of template and hook methods are the same
- References to H-objects are managed by AddH() and RemoveH()

UNIVERSITÄT SALZBURG

Example: Definition of an Object Hierarchy

```
T root= new T();
T subRoot= null;
root.AddH(new H2());
subRoot= new T();
root.AddH(subRoot);
root.AddH(new H1());
subRoot.AddH(new T());
subRoot.AddH(new H2());
```



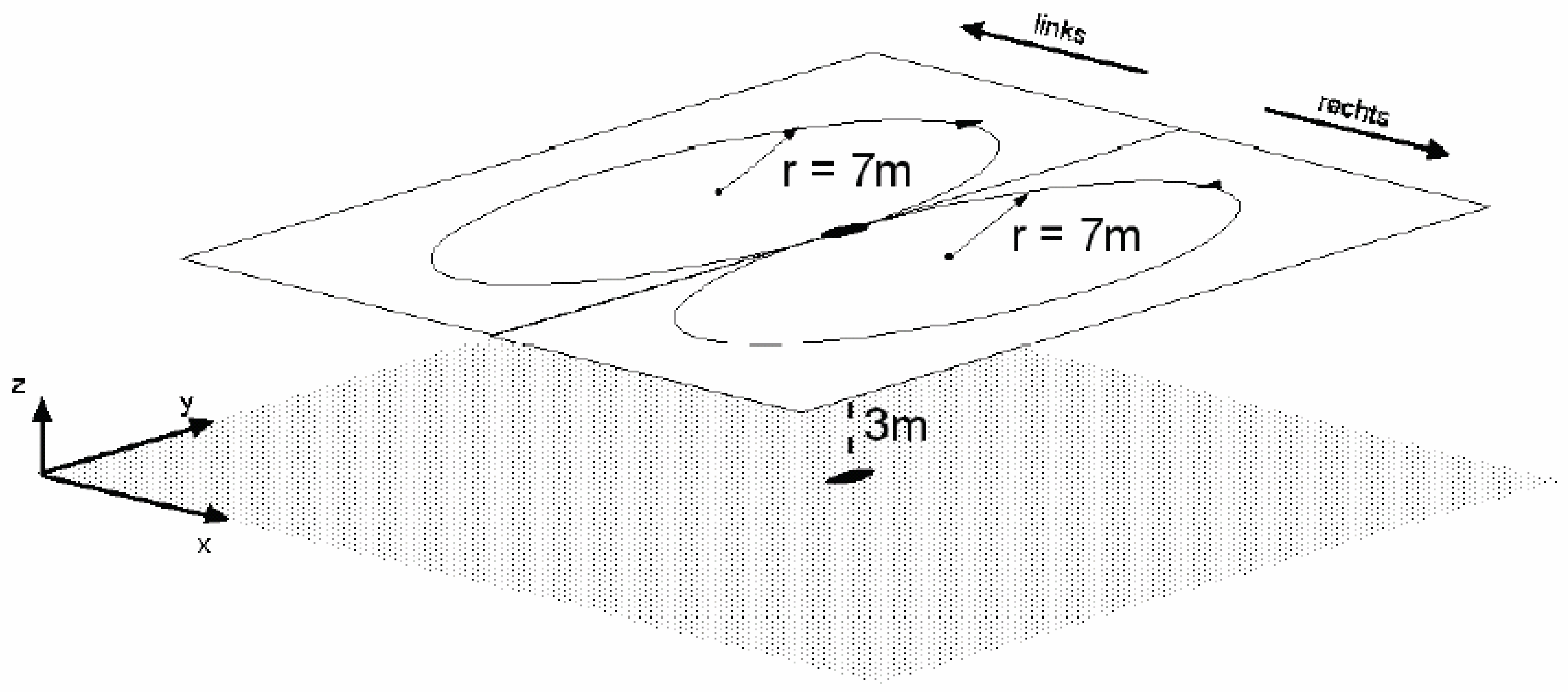
UNIVERSITÄT SALZBURG

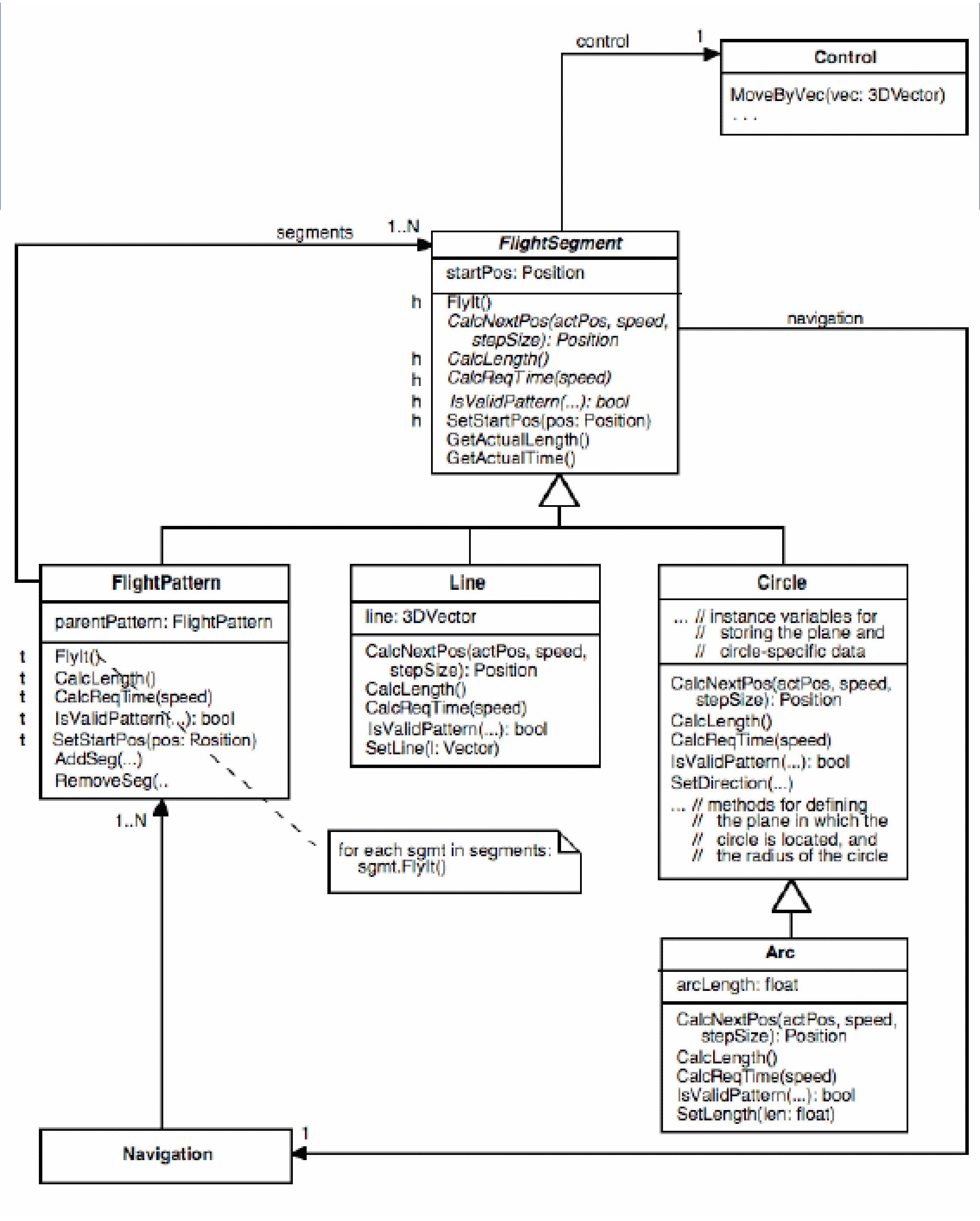
The object hierarchy can be used by the structure of the template method like an object

```
void M() {
     for each hObj in hList
          hObj.M();
  }
```

M () is not a recursive method, however it operates on a recursive data structure (tree).

Example: Composition of an 8-flight Pattern From Segments

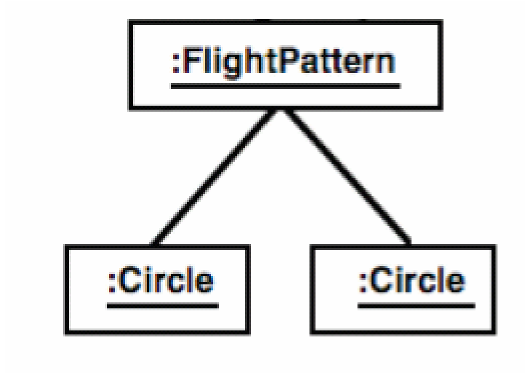



control
1
Control
MoveByVec(vec: 3DVector)
. . .
segments
1..N
FlightSegment
startPos: Position
h FlyIt()
CalcNextPos(actPos, speed, stepSize): Position
h CalcLength()
h CalcReqTime(speed)
h IsValidPattern(...): bool
h SetStartPos(pos: Position)
GetActualLength()
GetActualTime()
navigation
FlightPattern
parentPattern: FlightPattern
t FlyIt()
t CalcLength()
t CalcReqTime(speed)
t IsValidPattern(...): bool
t SetStartPos(pos: Position)
AddSeg(...)
RemoveSeg(..
Line
line: 3DVector
CalcNextPos(actPos, speed, stepSize): Position
CalcLength()
CalcReqTime(speed)
IsValidPattern(...): bool
SetLine(l: Vector)
Circle
... // instance variables for // storing the plane and // circle-specific data
CalcNextPos(actPos, speed, stepSize): Position
CalcLength()
CalcReqTime(speed)
IsValidPattern(...): bool
SetDirection(...)
... // methods for defining // the plane in which the // circle is located, and // the radius of the circle
1..N
for each sgmt in segments: sgmt.FlyIt()
Arc
arcLength: float
CalcNextPos(actPos, speed, stepSize): Position
CalcLength()
CalcReqTime(speed)
IsValidPattern(...): bool
SetLength(len: float)
Navigation
1

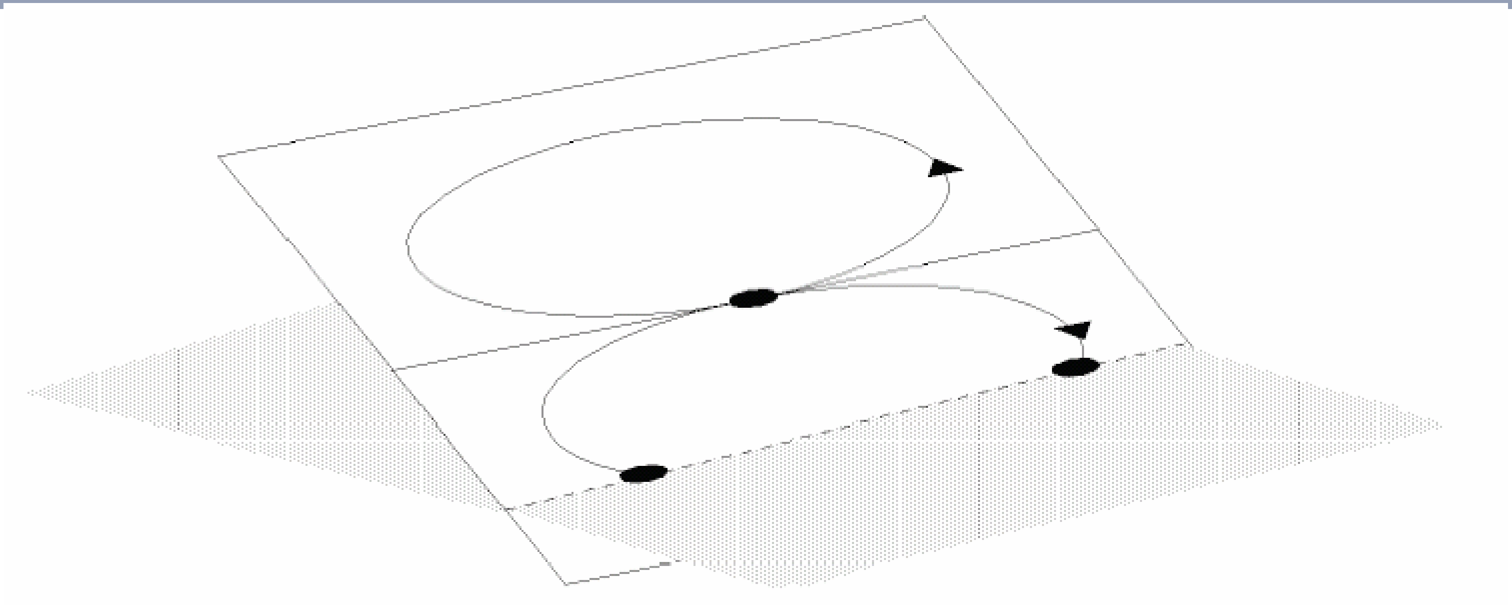
UNIVERSITÄT SALZBURG

The 8-loop

```
FlightPattern loop= new FlightPattern();
loop.SetStartPos(new Position(gL, gB) + new Position(0, 0, 3));
loop.AddSeg(new Circle (horizontalPlane, 7, right));  // radius: 7 m; right dir.
loop.AddSeg(new Circle (horizontalPlane, 7, left));  // radius: 7 m; left dir.
```



IsValidPattern() cheks whether a flight pattern leads to a ground contact



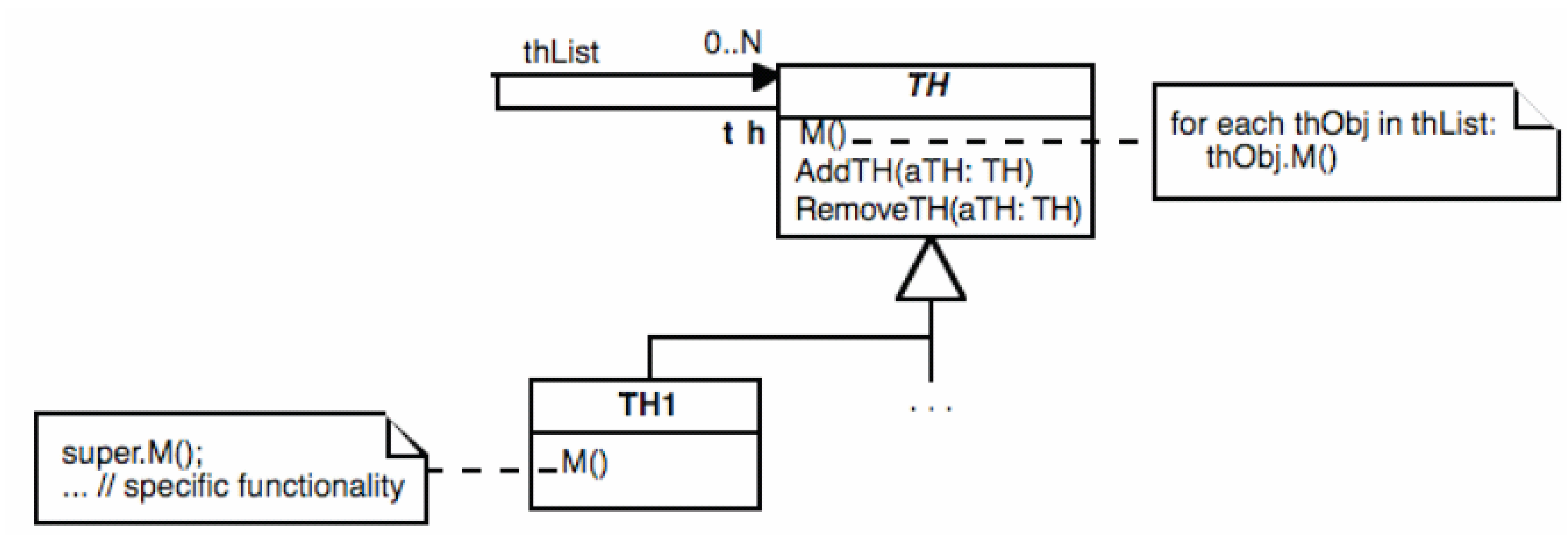
- IsValidPattern () is implemented in FlightPattern in accordance with the Composite template method
- Similarly: FlyIt (), CalcLength (), CalcReqTime ()
- FlyIt () is already implemented by using

FlightSegment - > CalcNextPos ()

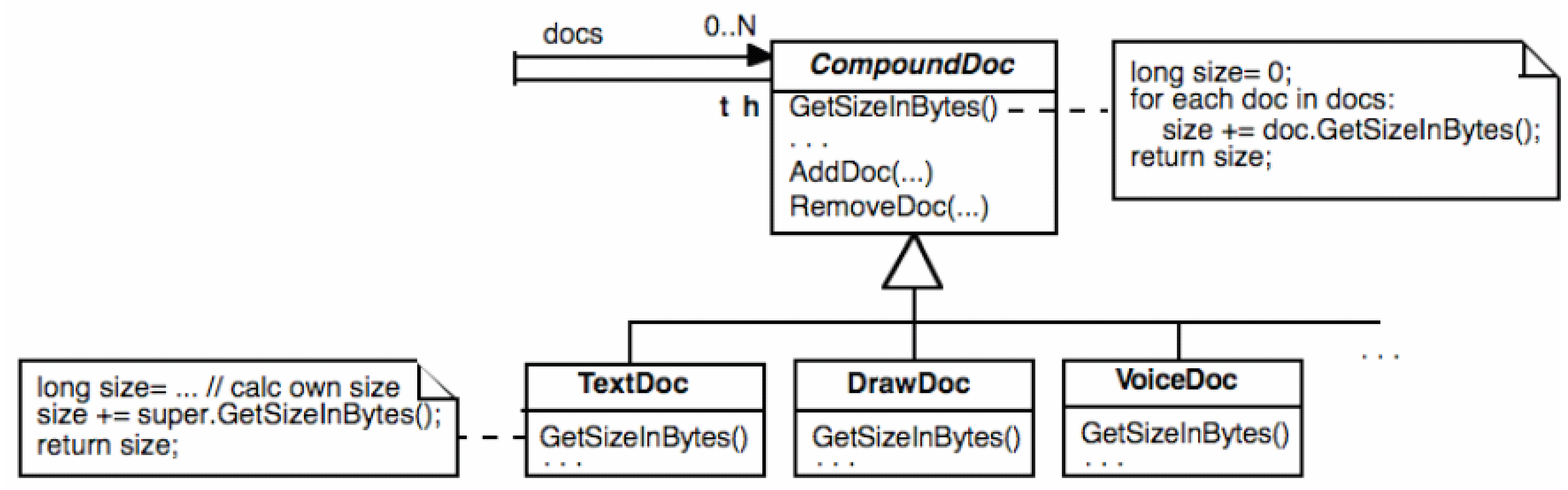
UNIVERSITÄT SALZBURG

Composite Variant: Administration and Functionality in One Class

- T and H class merged
- Semantics of the composition changes
- The fundamental characteristic to be able to define an object hierarchy remains



Example: Complex Documents



```
long size= 0;
for each doc in docs:
   size += doc.GetSizeInBytes();
return size;
```

```
long size= ... // calc own size
size += super.GetSizeInBytes();
return size;
```

A text document that comprises different other documents like drawings, audio or video clips, is responsible for the administration of the contained documents and offers additional functionality for editing the embedded documents.

UNIVERSITÄT SALZBURG

Summary *Composite*

+ Simple formation of flexible object hierarchies

+ New elements (subclasses of the hook class) without change of the template class

- Complexity of interactions between objects arranged in the hierarchy, in order to accomplish the automatic iteration over the tree hierarchy.

Object hierarchies occur very frequently and in many ranges of application, e.g. in window–grouped GUI elements, parts lists, workflows.