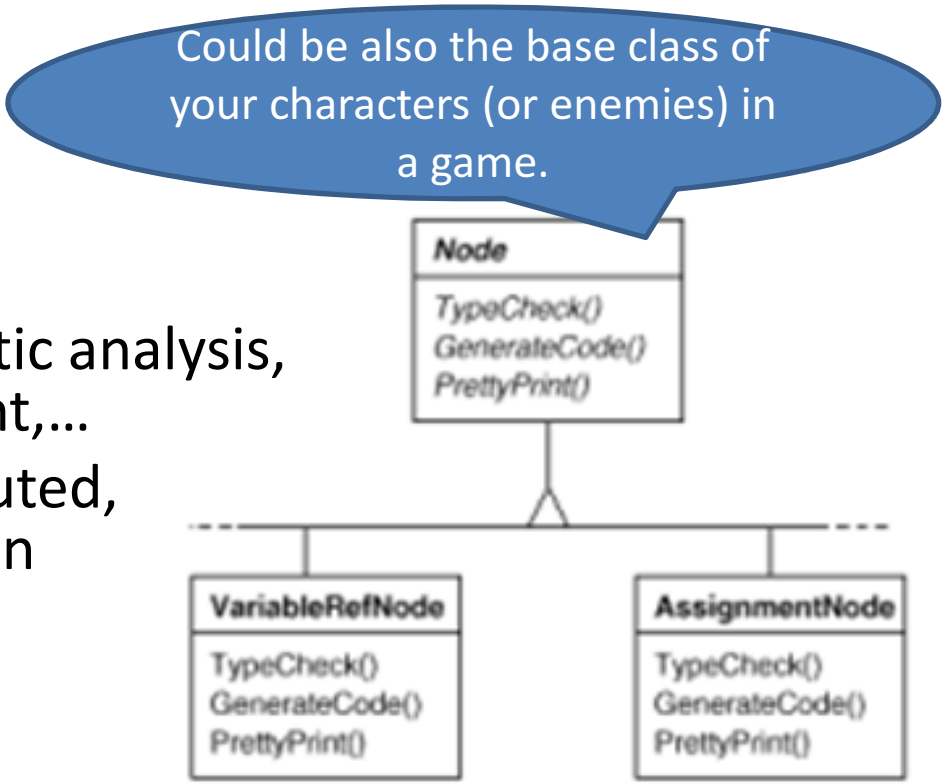
VISITOR PATTERN

Zweck & Motivation

- Ermöglicht es, **neue Operationen** auf den Elementen einer Struktur zu definieren, **ohne die Elemente selbst anzupassen**.

- Example: Compiler
  - Abstract syntax tree (AST)
  - Operations for static semantic analysis, code generation, pretty-print,…
  - → different aspects, distributed, hard to understand, maintain
  - introduce a new op?
    → Recompile all



Kontext

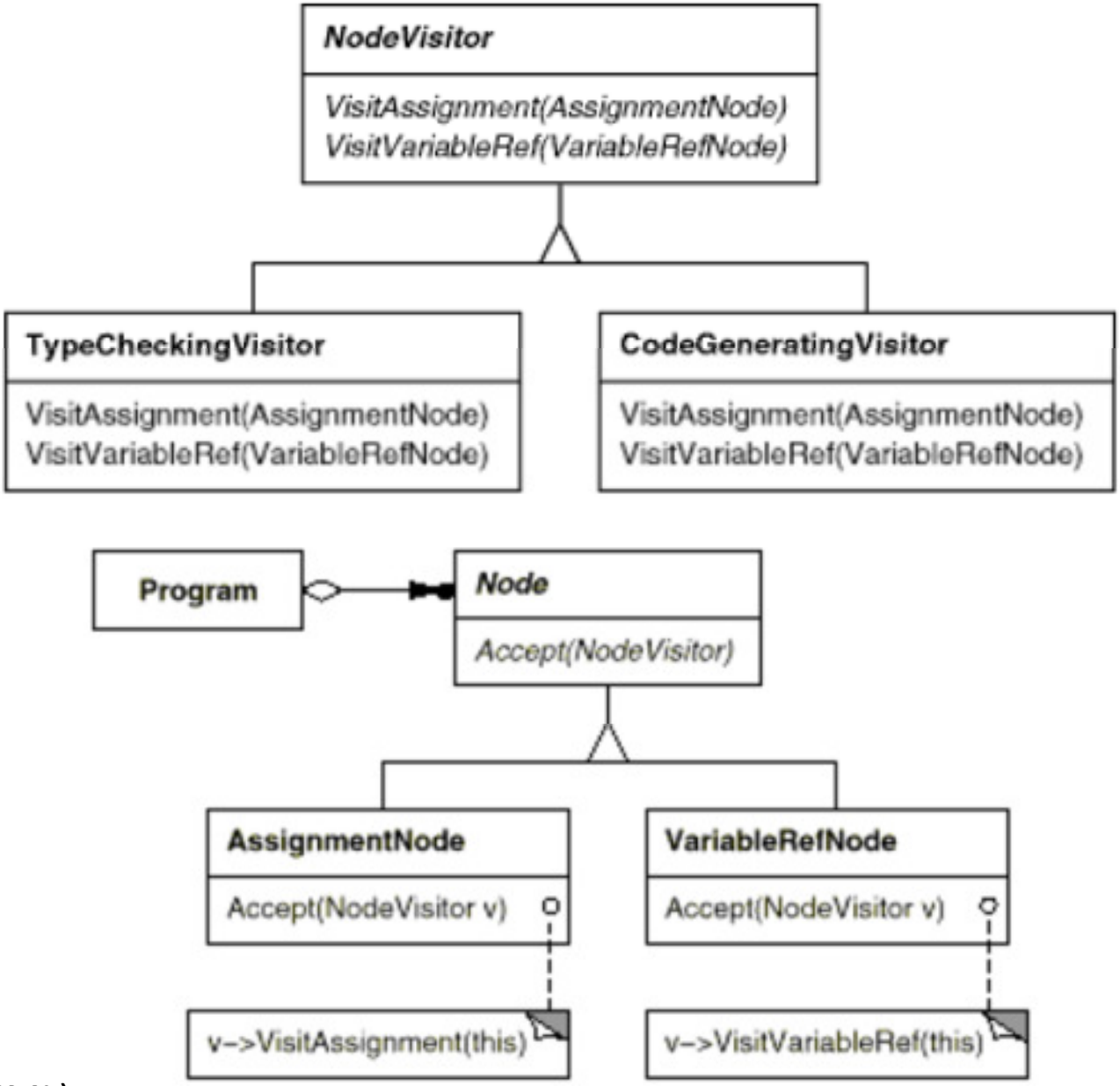
- Auf den unterschiedlichen Elementen einer Objektstruktur (z. B. Hierarchie bzw. Liste) sollen Operationen ermöglicht werden, die stark von den individuellen Eigenschaften der konkreten Elemente abhängen.
- Die Klassen der **Elemente** sollen dabei **nicht durch Methoden aufgebläht**, die verschiedenen Zwecken dienen und gegebenenfalls gar nicht in einem gemeinsamen Kontext zum Einsatz kommen.

Lösung

- Nicht Elemente der Struktur erweitern
- **Benötigte Operation** in einer neuen Klasse **kapseln (Visitor)**; diese arbeitet dann auf den Elementen der Struktur.

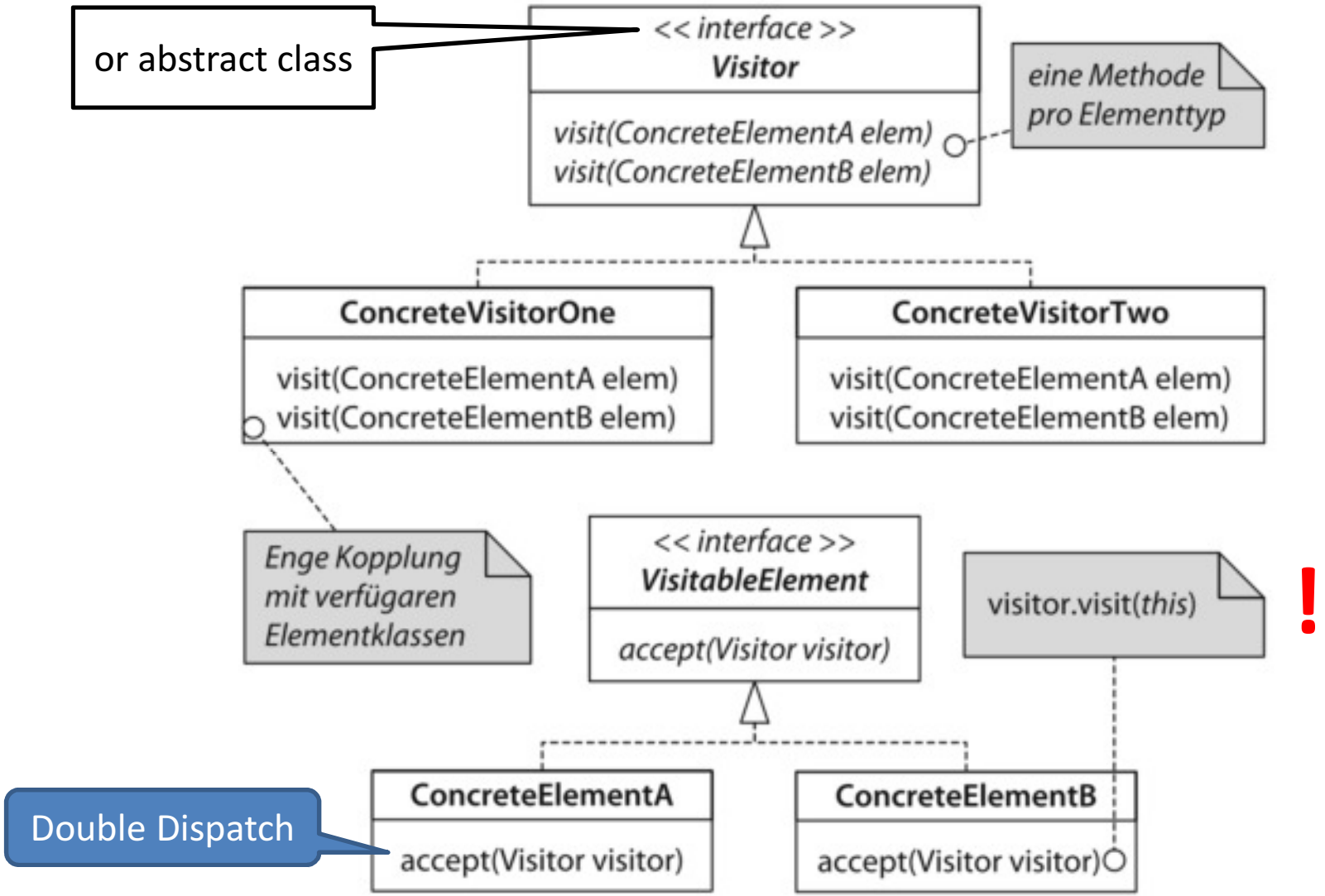
- Bei der Traversierung der Struktur wird den Elementen der Visitor übergeben. Wenn ein Element den Visitor „akzeptiert“, ruft es die Operation des Visitors auf, die dem Element entspricht, und übergibt *this* als Parameter.

- z.B. TypeCheckingVisitor bietet visitAssignment, visitVariableRef, etc.


NodeVisitor
VisitAssignment(AssignmentNode)
VisitVariableRef(VariableRefNode)
TypeCheckingVisitor
VisitAssignment(AssignmentNode)
VisitVariableRef(VariableRefNode)
CodeGeneratingVisitor
VisitAssignment(AssignmentNode)
VisitVariableRef(VariableRefNode)
Program
Node
Accept(NodeVisitor)
AssignmentNode
Accept(NodeVisitor v)
VariableRefNode
Accept(NodeVisitor v)
v->VisitAssignment(this)
v->VisitVariableRef(this)

[GOF] (Not UML)

Visitor Pattern



Participants

- Visitor: declares a Visit operation for each class of ConcreteElement in the object structure. The operation's name and signature identifies the class that sends the Visit request to the visitor. That lets the visitor determine the concrete class of the element being visited.
- ConcreteVisitor: implements each operation declared by Visitor. Each operation implements a fragment of the algorithm defined for the corresponding class of object in the structure.
- VisitableElement : defines an Accept operation that takes a visitor as an argument.
- ConcreteElement: implements an Accept operation that takes a visitor as an argument.

Advantages

- Visitor makes adding new operations easy.
  - You can define a new operation over an object structure simply by adding a new visitor.
    In contrast, if you spread functionality over many classes, then you must change each class to define a new operation.
- A visitor gathers related operations and separates unrelated ones. → localized
- Visiting across class hierarchies.
  - It can visit objects that don't have a common parent class. (unlike Iterator)
- Accumulating state
  - Visitors can accumulate state as they visit each element in the object structure. Otherwise: state would be passed as extra arguments to the operations that perform the traversal, or they might appear as global variables.

- Disadvantages
  - Tight coupling (Visitor – Elements)
  - Adding new ConcreteElement classes is hard.
    - 1) change interface (new abstract operation)
    - 2) change all concrete visitors.
  - Breaking encapsulation.
    - the pattern often forces you to provide public operations that access an element's internal state.

- Traversal
  - In object structure, visitor, or separate iterator.

Verwendung

- Eine Objektstruktur enthält **Elemente unterschiedlicher Klassen**.
- Die Menge der **Elementtypen** (ConcreteElements) ist abgeschlossen oder **ändert sich nur sehr selten**.
- Es sollen Operationen auf den Elementen der Struktur implementiert werden, die stark von den individuellen Objekteigenschaften abhängen.
- Die Methoden können oder sollen aufgrund ihrer Anzahl, ihres **unterschiedlichen Einsatzzwecks** oder aber bedingt durch die Rollenverteilung **nicht in den Interfaces der Elemente definiert** werden.

Double-Dispatch

- See http://www.codeproject.com/Articles/588882/TheplusVisitorplusPatternplusExplained