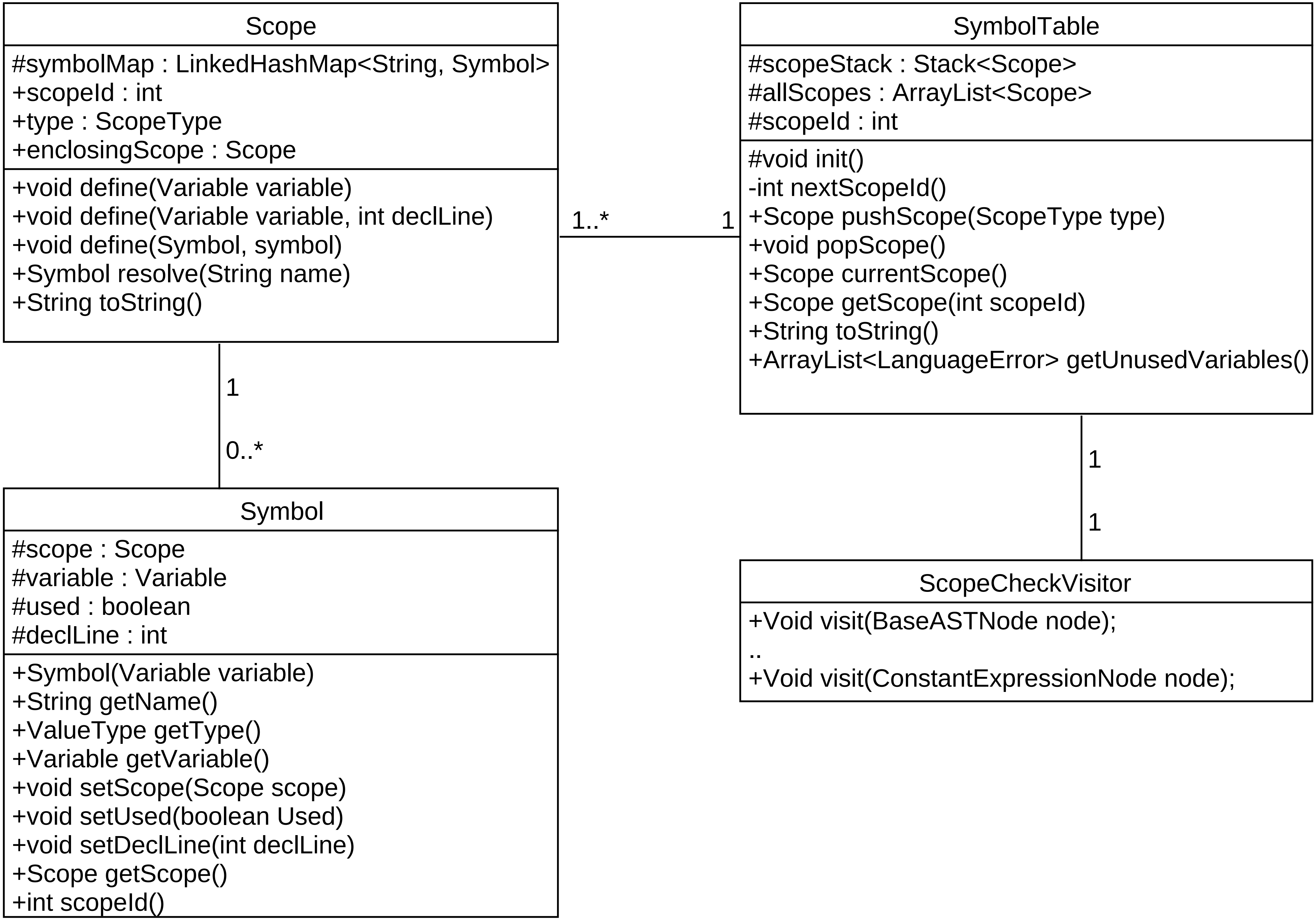

Scope

#symbolMap : LinkedHashMap<String, Symbol>
+scopeId : int
+type : ScopeType
+enclosingScope : Scope

+void define(Variable variable)
+void define(Variable variable, int declLine)
+void define(Symbol, symbol)
+Symbol resolve(String name)
+String toString()

SymbolTable

#scopeStack : Stack<Scope>
#allScopes : ArrayList<Scope>
#scopeId : int

#void init()
-int nextScopeId()
+Scope pushScope(ScopeType type)
+void popScope()
+Scope currentScope()
+Scope getScope(int scopeId)
+String toString()
+ArrayList<LanguageError> getUnusedVariables()

Symbol

#scope : Scope
#variable : Variable
#used : boolean
#declLine : int

+Symbol(Variable variable)
+String getName()
+ValueType getType()
+Variable getVariable()
+void setScope(Scope scope)
+void setUsed(boolean Used)
+void setDeclLine(int declLine)
+Scope getScope()
+int scopeId()

ScopeCheckVisitor

+Void visit(BaseASTNode node);
..
+Void visit(ConstantExpressionNode node);