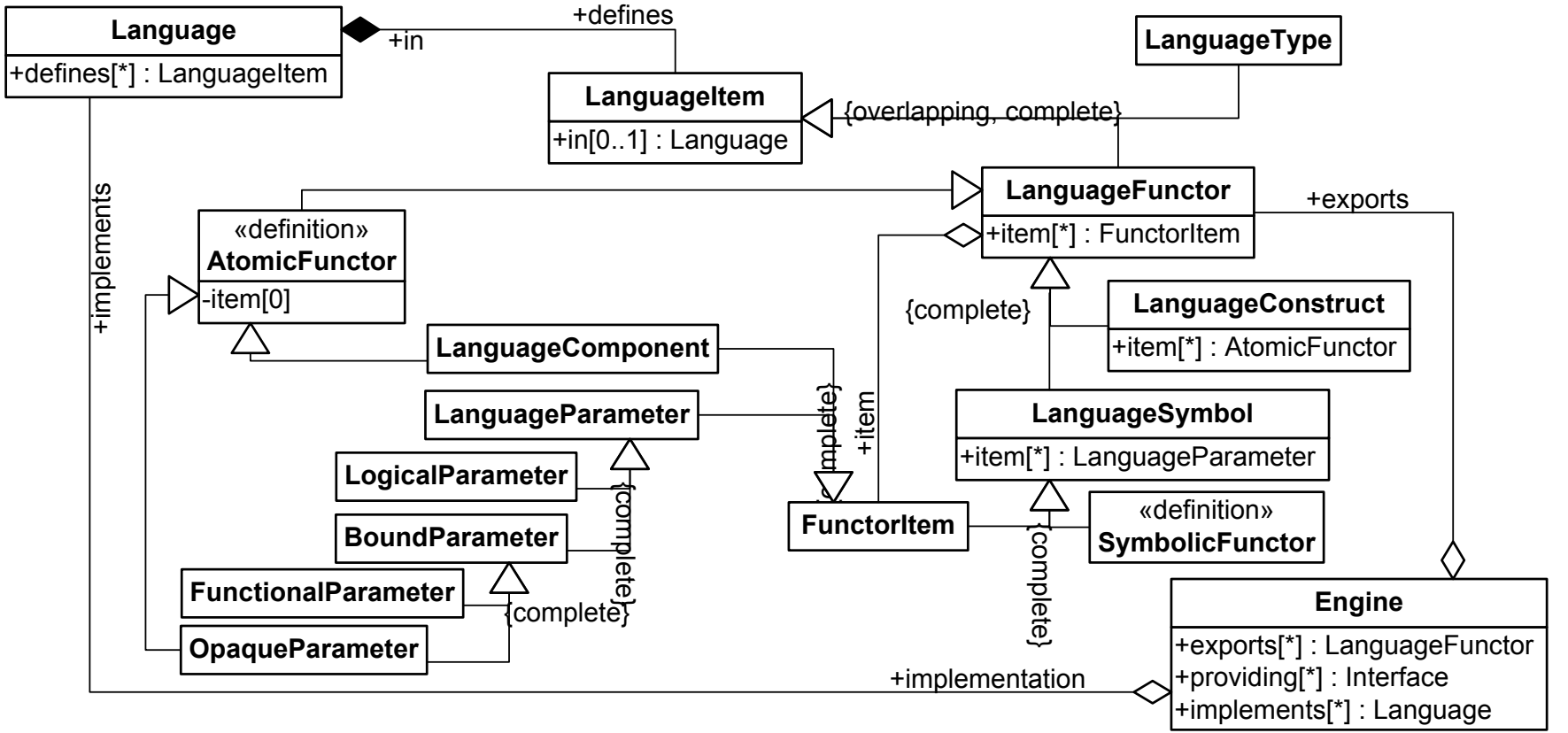
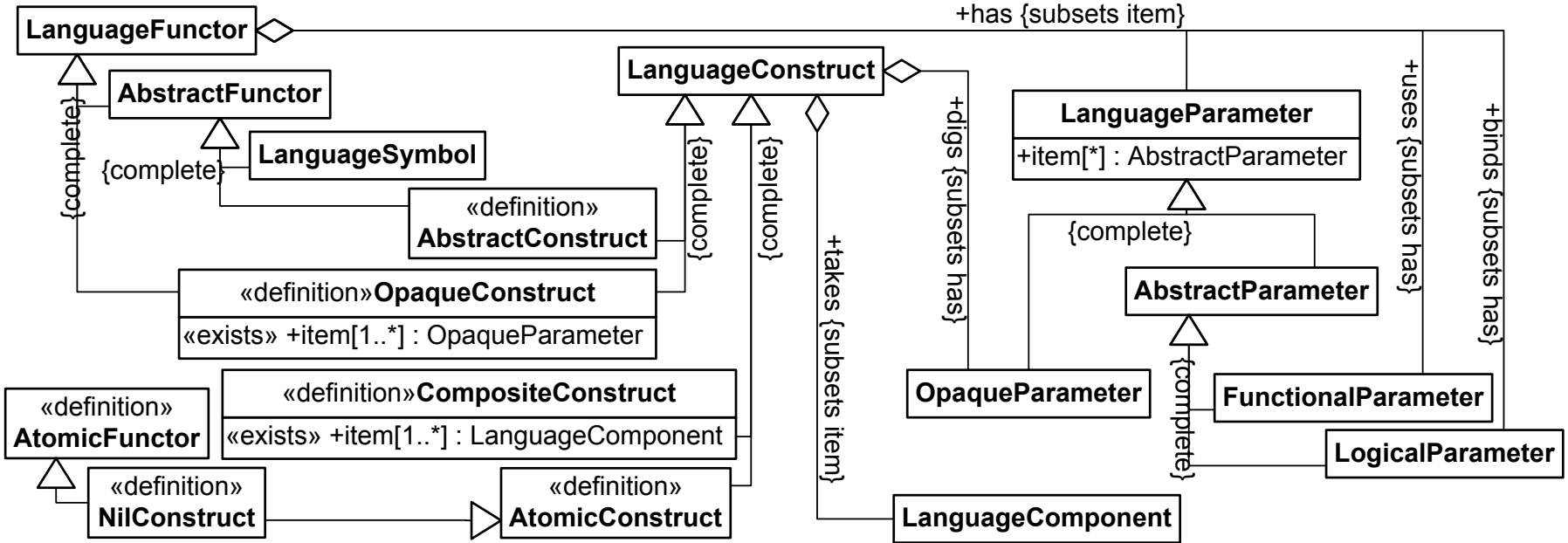
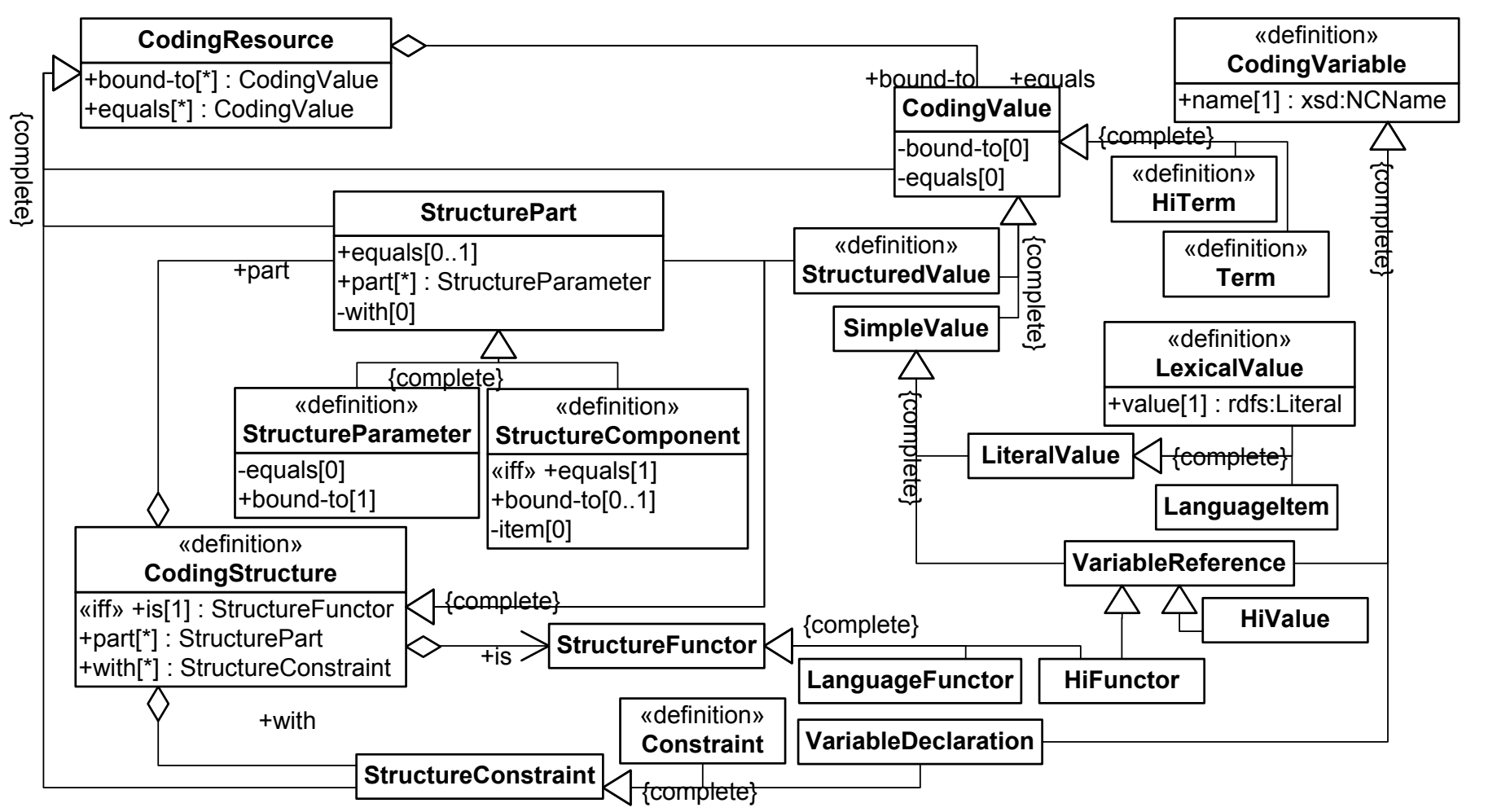
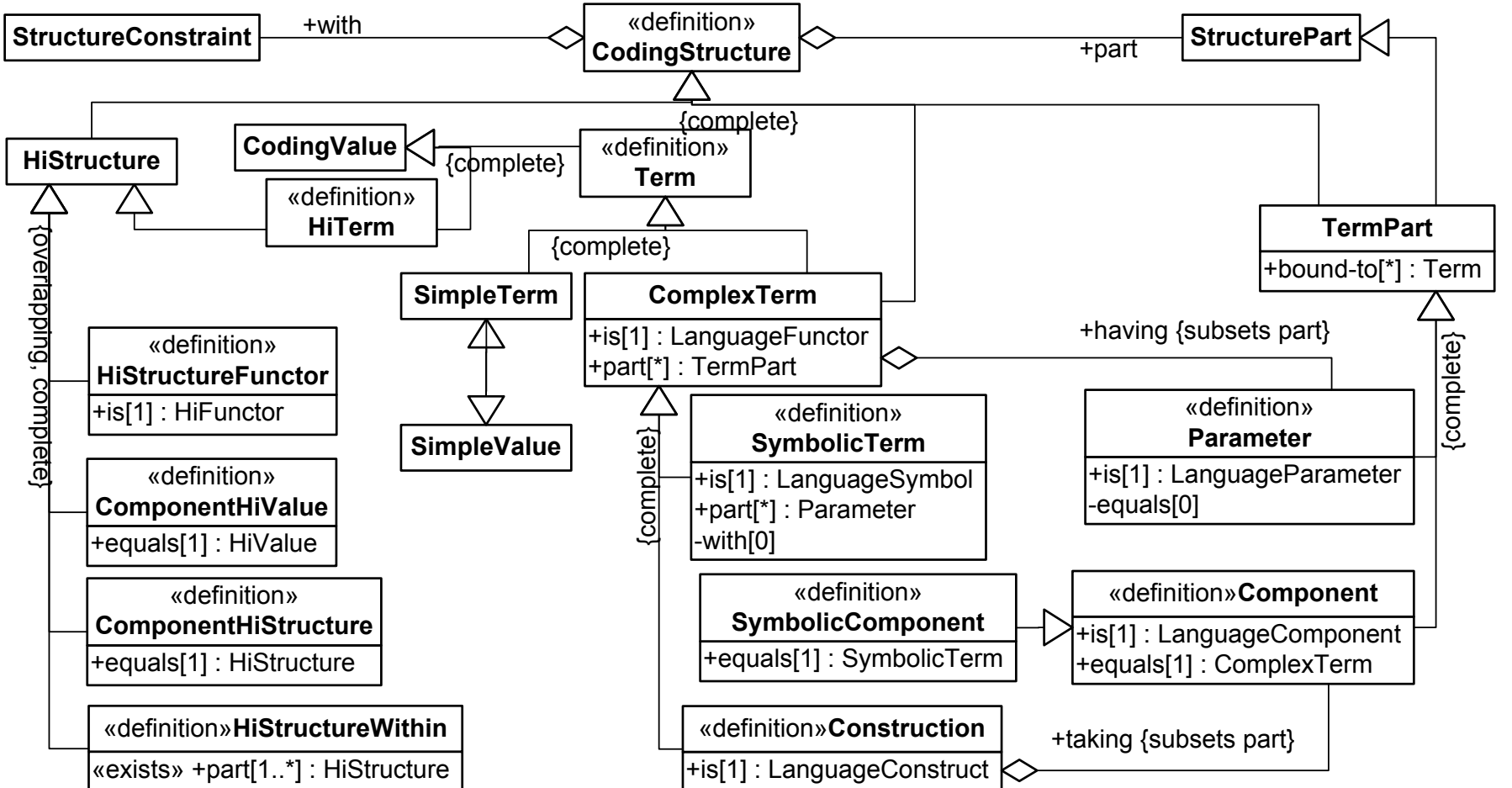
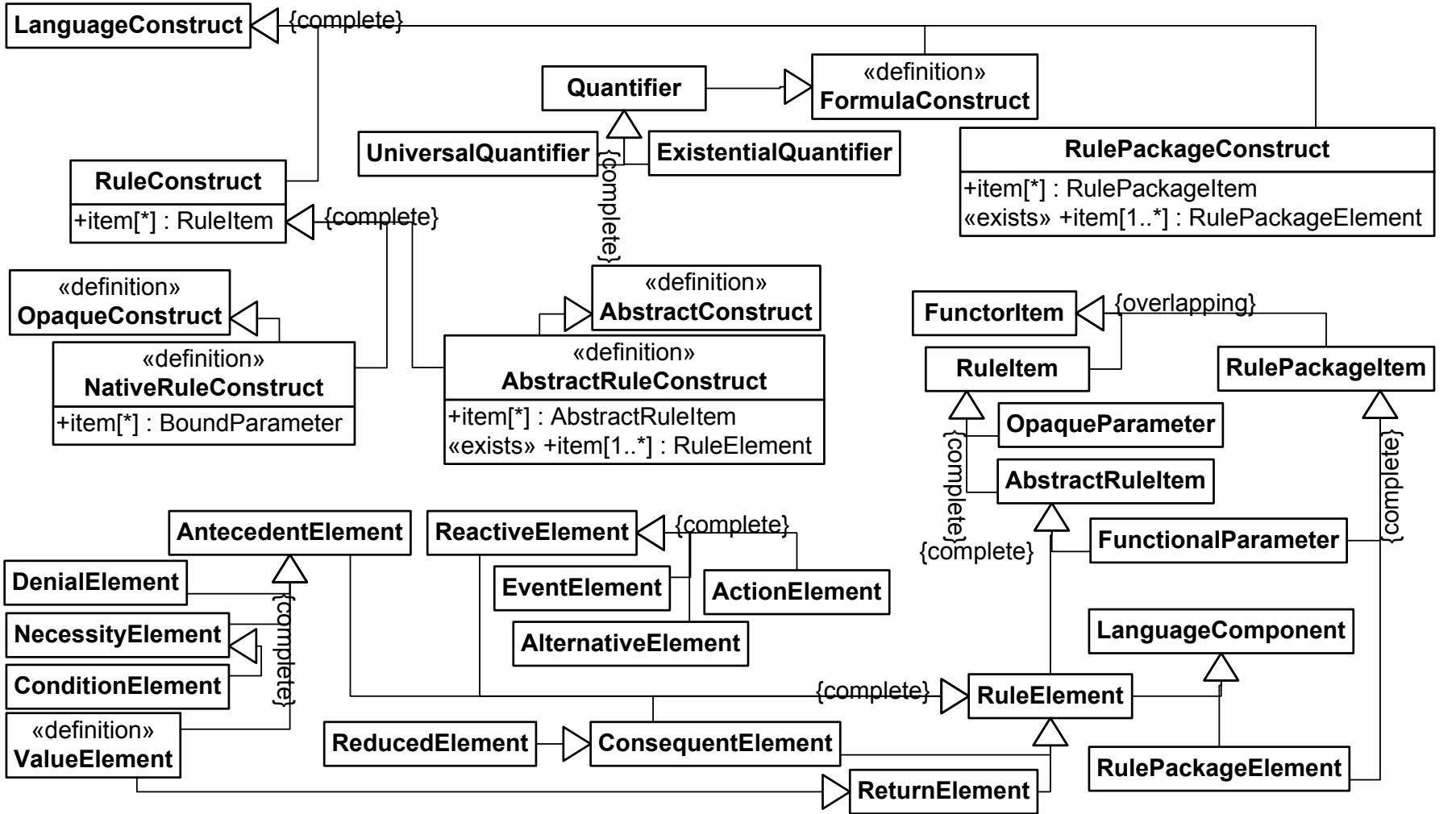
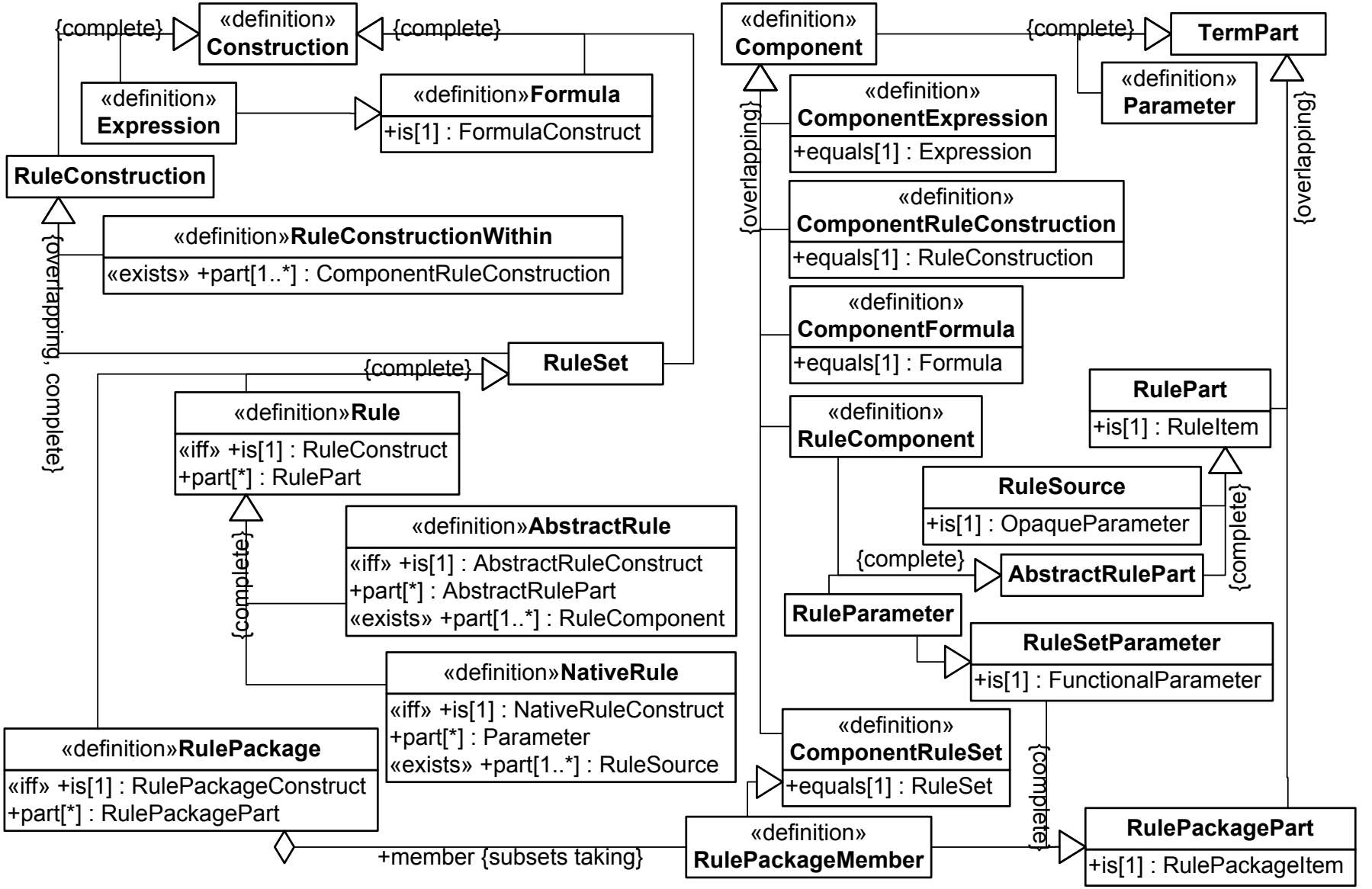

Language
+defines[*] : LanguageItem
+in
+defines
LanguageItem
+in[0..1] : Language
LanguageType
{overlapping, complete}
LanguageFunctor
+item[*] : FunctorItem
+exports
+implements
«definition»
AtomicFunctor
-item[0]
{complete}
LanguageConstruct
+item[*] : AtomicFunctor
LanguageComponent
LanguageParameter
+item
LanguageSymbol
+item[*] : LanguageParameter
LogicalParameter
BoundParameter
FunctionalParameter
OpaqueParameter
{complete}
FunctorItem
«definition»
SymbolicFunctor
Engine
+exports[*] : LanguageFunctor
+providing[*] : Interface
+implements[*] : Language
+implementation

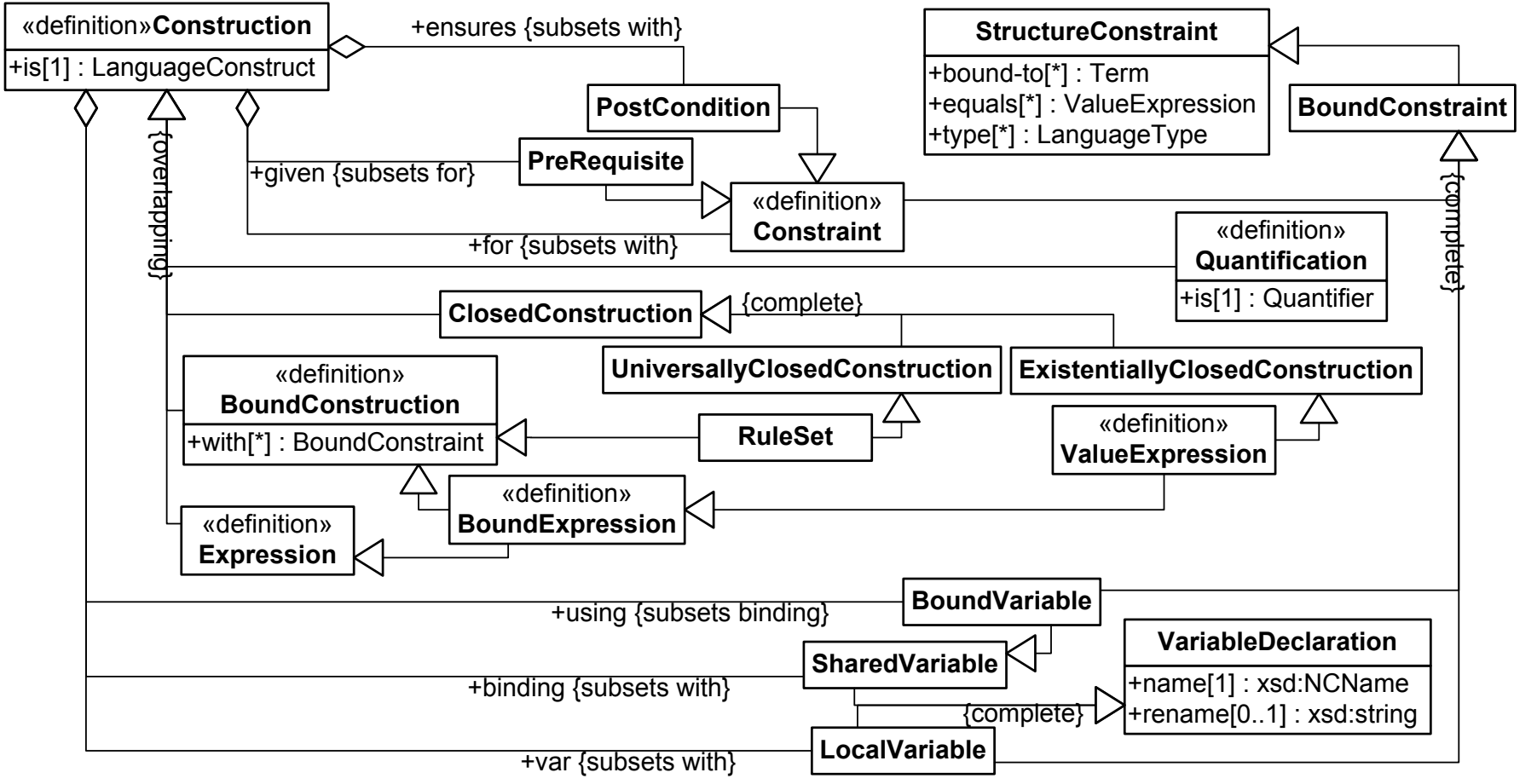

LanguageFunctor
+has {subsets item}
AbstractFunctor
{complete}
LanguageSymbol
{complete}
«definition»
AbstractConstruct
«definition»OpaqueConstruct
«exists» +item[1..*] : OpaqueParameter
«definition»
AtomicFunctor
«definition»
NilConstruct
«definition»CompositeConstruct
«exists» +item[1..*] : LanguageComponent
«definition»
AtomicConstruct
LanguageConstruct
{complete}
{complete}
+takes {subsets item}
+digs {subsets has}
LanguageParameter
+item[*] : AbstractParameter
{complete}
AbstractParameter
OpaqueParameter
{complete}
FunctionalParameter
LogicalParameter
+uses {subsets has}
+binds {subsets has}
LanguageComponent

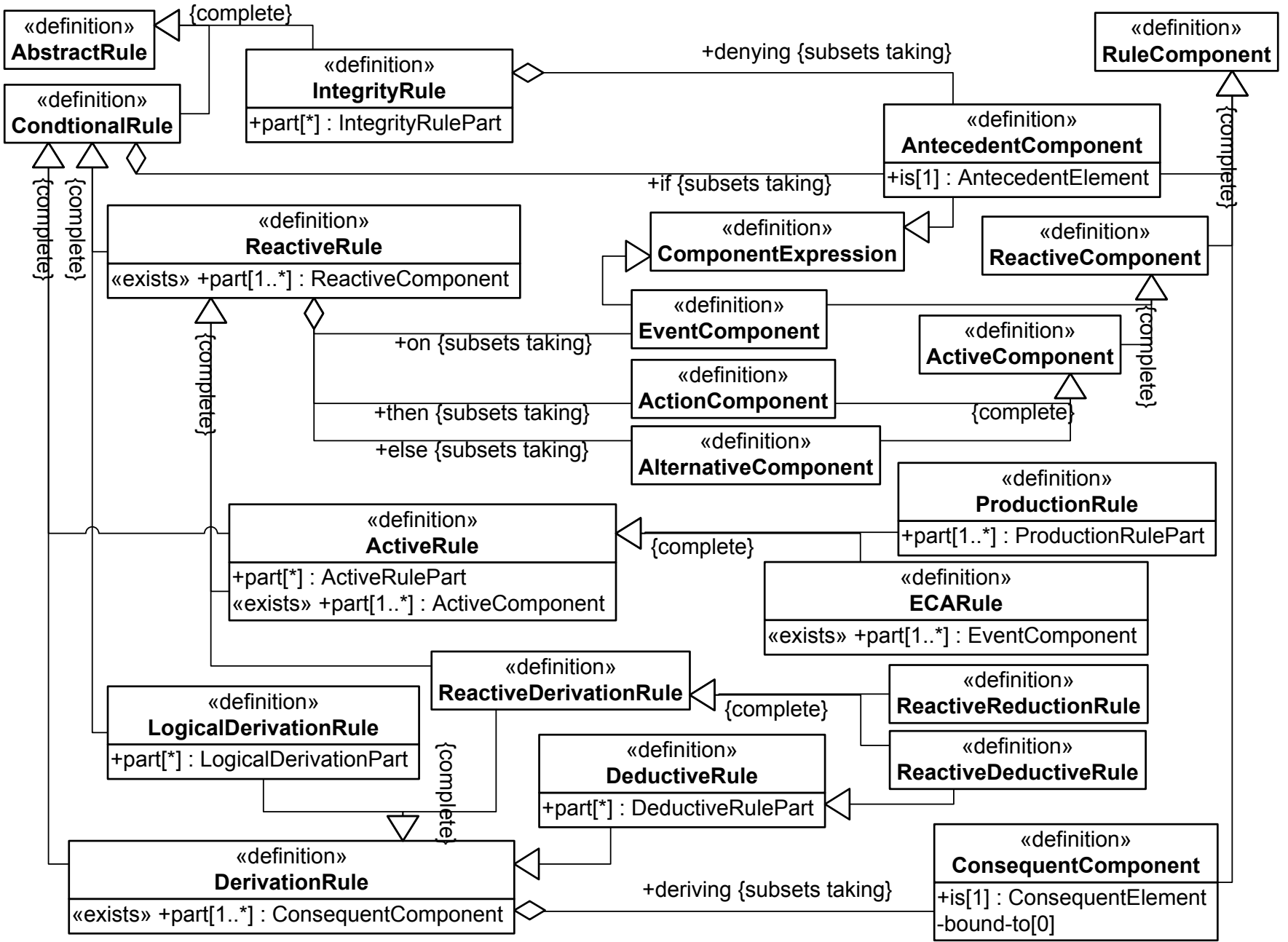

CodingResource
+bound-to[*] : CodingValue
+equals[*] : CodingValue
+bound-to
+equals
CodingValue
-bound-to[0]
-equals[0]
«definition»
CodingVariable
+name[1] : xsd:NCName
{complete}
«definition»
HiTerm
«definition»
Term
StructurePart
+equals[0..1]
+part[*] : StructureParameter
-with[0]
+part
«definition»
StructuredValue
SimpleValue
«definition»
LexicalValue
+value[1] : rdfs:Literal
LiteralValue
LanguageItem
«definition»
StructureParameter
-equals[0]
+bound-to[1]
«definition»
StructureComponent
«iff» +equals[1]
+bound-to[0..1]
-item[0]
«definition»
CodingStructure
«iff» +is[1] : StructureFunctor
+part[*] : StructurePart
+with[*] : StructureConstraint
+is
StructureFunctor
VariableReference
HiValue
LanguageFunctor
HiFunctor
+with
«definition»
Constraint
VariableDeclaration
StructureConstraint

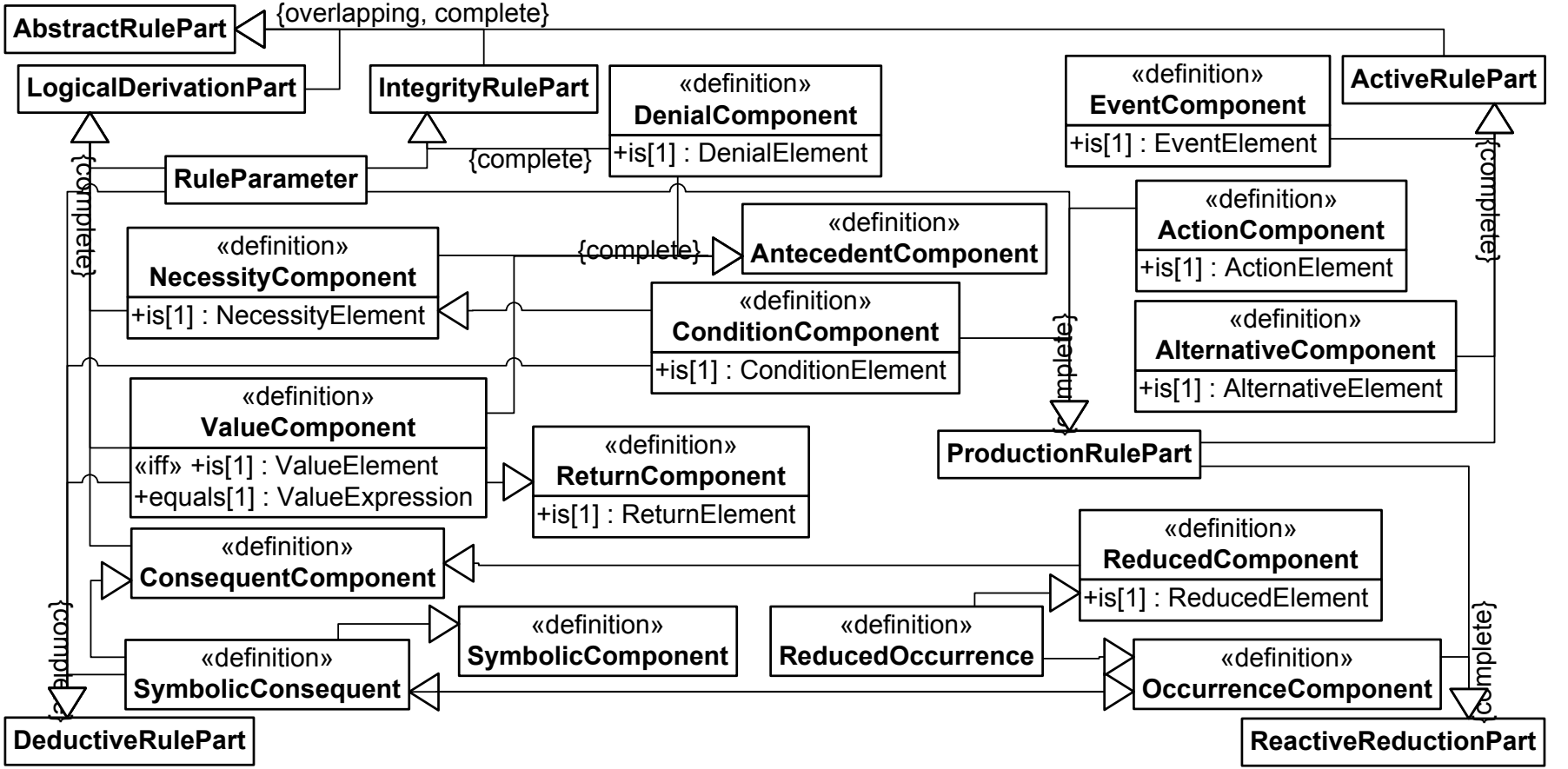

StructureConstraint
+with
«definition»
CodingStructure
+part
StructurePart
{complete}
HiStructure
CodingValue
{complete}
«definition»
Term
«definition»
HiTerm
{complete}
TermPart
+bound-to[*] : Term
{overlapping, complete}
SimpleTerm
ComplexTerm
+is[1] : LanguageFunctor
+part[*] : TermPart
+having {subsets part}
«definition»
HiStructureFunctor
+is[1] : HiFunctor
SimpleValue
«definition»
SymbolicTerm
+is[1] : LanguageSymbol
+part[*] : Parameter
-with[0]
«definition»
Parameter
+is[1] : LanguageParameter
-equals[0]
{complete}
{complete}
«definition»
ComponentHiValue
+equals[1] : HiValue
«definition»
SymbolicComponent
+equals[1] : SymbolicTerm
«definition»Component
+is[1] : LanguageComponent
+equals[1] : ComplexTerm
«definition»
ComponentHiStructure
+equals[1] : HiStructure
«definition»HiStructureWithin
«exists» +part[1..*] : HiStructure
«definition»Construction
+is[1] : LanguageConstruct
+taking {subsets part}

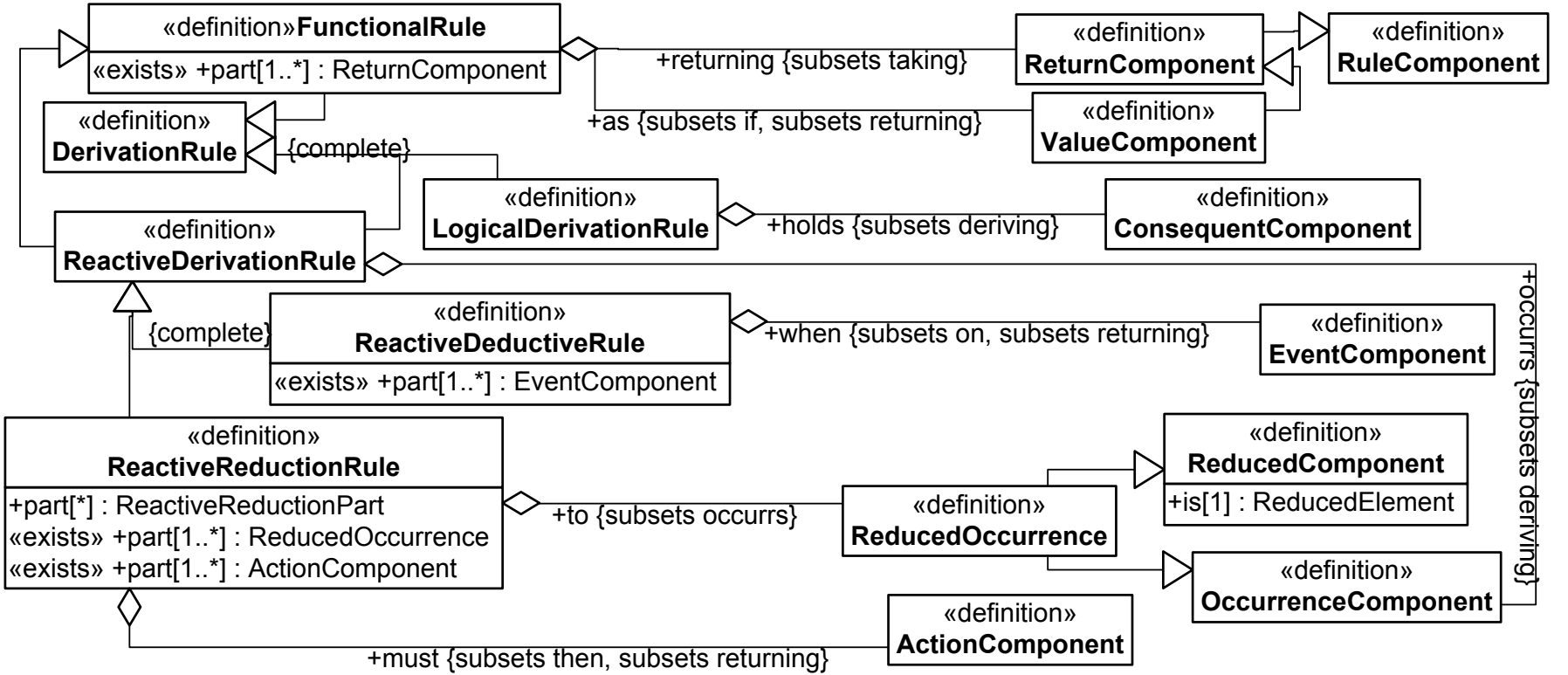

LanguageConstruct
{complete}
Quantifier
«definition»
FormulaConstruct
UniversalQuantifier
{complete}
ExistentialQuantifier
RulePackageConstruct
+item[*] : RulePackageItem
«exists» +item[1..*] : RulePackageElement
RuleConstruct
+item[*] : RuleItem
{complete}
«definition»
OpaqueConstruct
«definition»
AbstractConstruct
«definition»
NativeRuleConstruct
+item[*] : BoundParameter
«definition»
AbstractRuleConstruct
+item[*] : AbstractRuleItem
«exists» +item[1..*] : RuleElement
FunctorItem
{overlapping}
RuleItem
RulePackageItem
OpaqueParameter
{complete}
AbstractRuleItem
{complete}
{complete}
FunctionalParameter
{complete}
AntecedentElement
ReactiveElement
{complete}
DenialElement
EventElement
ActionElement
NecessityElement
AlternativeElement
LanguageComponent
ConditionElement
{complete}
{complete}
RuleElement
«definition»
ValueElement
ReducedElement
ConsequentElement
RulePackageElement
ReturnElement

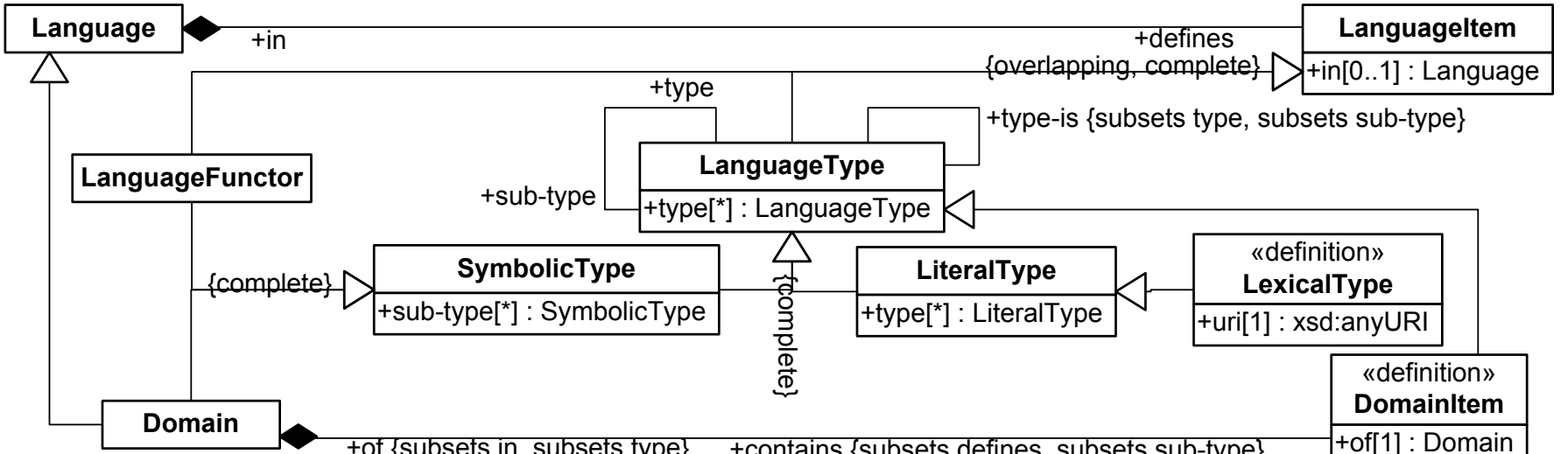

«definition» Construction
{complete}
{complete}
«definition» Expression
«definition»Formula
+is[1] : FormulaConstruct
RuleConstruction
«definition»RuleConstructionWithin
«exists» +part[1..*] : ComponentRuleConstruction
{overlapping, complete}
RuleSet
{complete}
«definition»Rule
«iff» +is[1] : RuleConstruct
+part[*] : RulePart
{complete}
«definition»AbstractRule
«iff» +is[1] : AbstractRuleConstruct
+part[*] : AbstractRulePart
«exists» +part[1..*] : RuleComponent
«definition»NativeRule
«iff» +is[1] : NativeRuleConstruct
+part[*] : Parameter
«exists» +part[1..*] : RuleSource
«definition»RulePackage
«iff» +is[1] : RulePackageConstruct
+part[*] : RulePackagePart
+member {subsets taking}
«definition» Component
{complete}
TermPart
«definition» Parameter
{overlapping}
{overlapping}
«definition» ComponentExpression
+equals[1] : Expression
«definition» ComponentRuleConstruction
+equals[1] : RuleConstruction
«definition» ComponentFormula
+equals[1] : Formula
«definition» RuleComponent
RulePart
+is[1] : RuleItem
RuleSource
+is[1] : OpaqueParameter
{complete}
{complete}
AbstractRulePart
RuleParameter
RuleSetParameter
+is[1] : FunctionalParameter
«definition» ComponentRuleSet
+equals[1] : RuleSet
{complete}
«definition» RulePackageMember
RulePackagePart
+is[1] : RulePackageItem

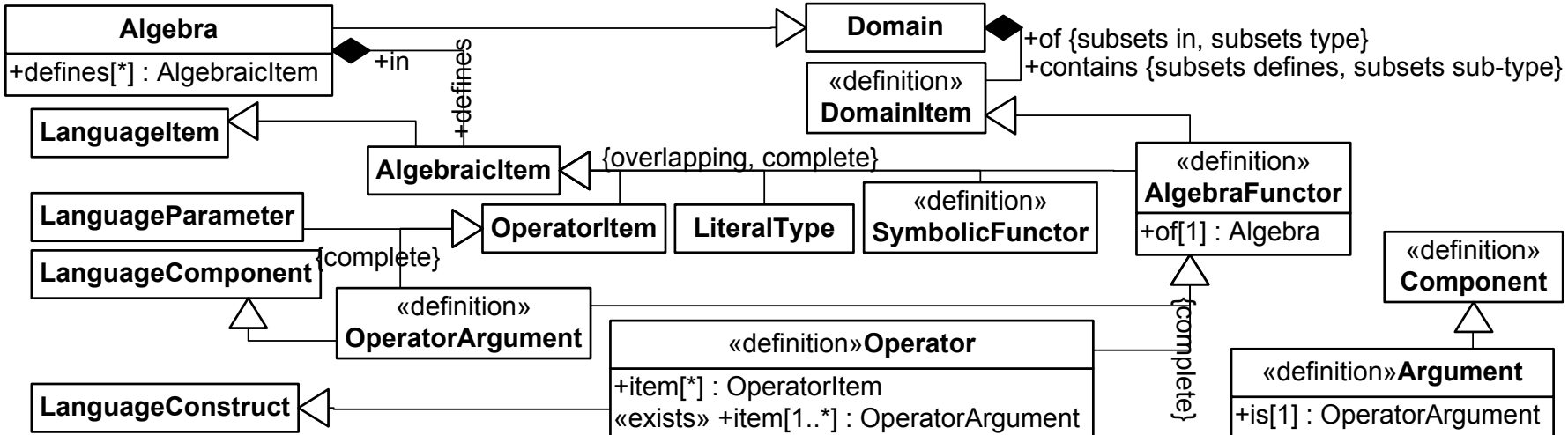

«definition»Construction
+is[1] : LanguageConstruct
+ensures {subsets with}
StructureConstraint
+bound-to[*] : Term
+equals[*] : ValueExpression
+type[*] : LanguageType
BoundConstraint
PostCondition
PreRequisite
+given {subsets for}
«definition»
Constraint
+for {subsets with}
{overlapping}
{complete}
«definition»
Quantification
+is[1] : Quantifier
ClosedConstruction
{complete}
UniversallyClosedConstruction
ExistentiallyClosedConstruction
«definition»
BoundConstruction
+with[*] : BoundConstraint
RuleSet
«definition»
ValueExpression
«definition»
BoundExpression
«definition»
Expression
BoundVariable
+using {subsets binding}
SharedVariable
VariableDeclaration
+name[1] : xsd:NCName
+rename[0..1] : xsd:string
+binding {subsets with}
{complete}
LocalVariable
+var {subsets with}

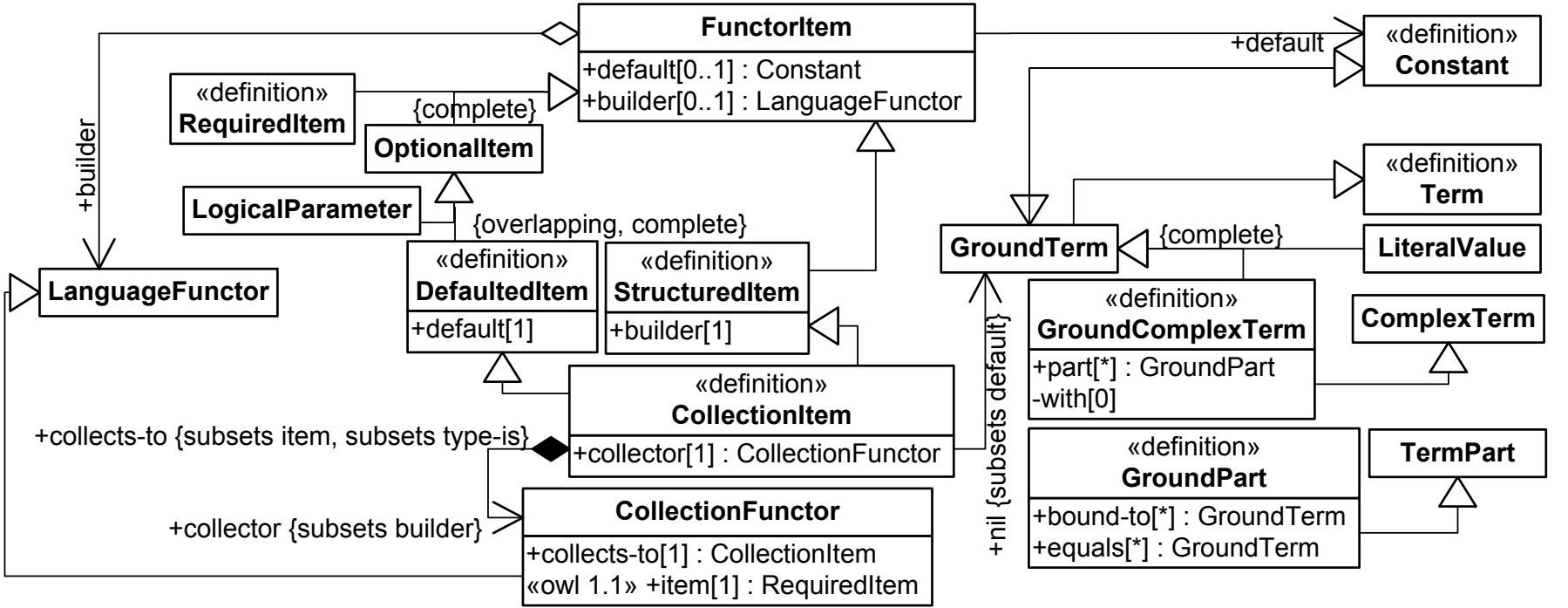

«definition»
AbstractRule
{complete}
«definition»
IntegrityRule
+part[*] : IntegrityRulePart
+denying {subsets taking}
«definition»
RuleComponent
«definition»
CondtionalRule
«definition»
AntecedentComponent
+is[1] : AntecedentElement
+if {subsets taking}
{complete}
{complete}
{complete}
«definition»
ReactiveRule
«exists» +part[1..*] : ReactiveComponent
«definition»
ComponentExpression
«definition»
ReactiveComponent
«definition»
EventComponent
+on {subsets taking}
«definition»
ActiveComponent
{complete}
{complete}
«definition»
ActionComponent
+then {subsets taking}
{complete}
+else {subsets taking}
«definition»
AlternativeComponent
«definition»
ProductionRule
+part[1..*] : ProductionRulePart
«definition»
ActiveRule
+part[*] : ActiveRulePart
«exists» +part[1..*] : ActiveComponent
{complete}
«definition»
ECARule
«exists» +part[1..*] : EventComponent
«definition»
ReactiveDerivationRule
{complete}
«definition»
ReactiveReductionRule
«definition»
LogicalDerivationRule
+part[*] : LogicalDerivationPart
«definition»
ReactiveDeductiveRule
«definition»
DeductiveRule
+part[*] : DeductiveRulePart
{complete}
«definition»
DerivationRule
«exists» +part[1..*] : ConsequentComponent
+deriving {subsets taking}
«definition»
ConsequentComponent
+is[1] : ConsequentElement
-bound-to[0]

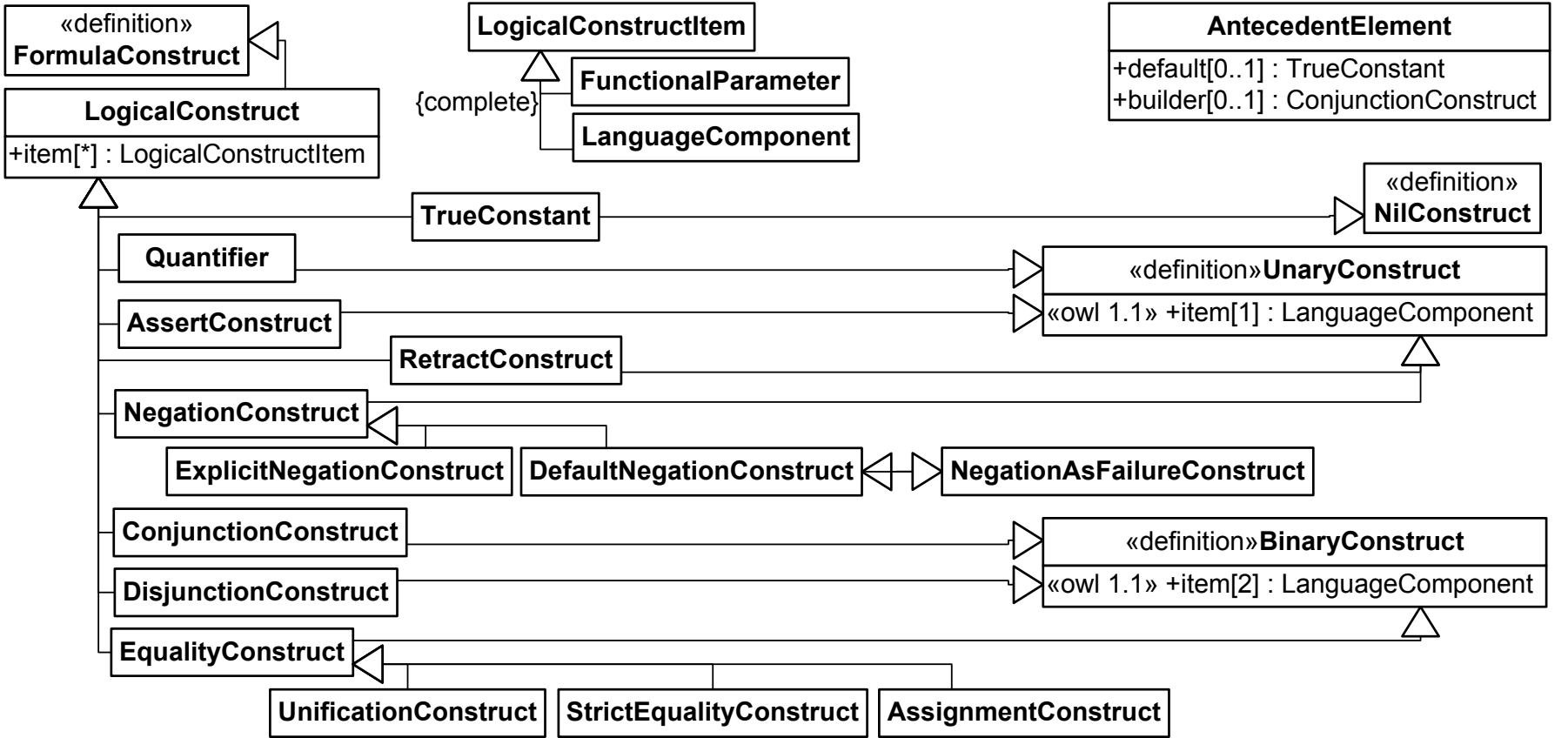

AbstractRulePart
{overlapping, complete}
LogicalDerivationPart
IntegrityRulePart
«definition»
DenialComponent
+is[1] : DenialElement
«definition»
EventComponent
+is[1] : EventElement
ActiveRulePart
RuleParameter
{complete}
«definition»
NecessityComponent
+is[1] : NecessityElement
«definition»
AntecedentComponent
«definition»
ActionComponent
+is[1] : ActionElement
«definition»
ConditionComponent
+is[1] : ConditionElement
«definition»
AlternativeComponent
+is[1] : AlternativeElement
«definition»
ValueComponent
«iff» +is[1] : ValueElement
+equals[1] : ValueExpression
«definition»
ReturnComponent
+is[1] : ReturnElement
ProductionRulePart
«definition»
ConsequentComponent
«definition»
ReducedComponent
+is[1] : ReducedElement
«definition»
SymbolicComponent
«definition»
ReducedOccurrence
«definition»
SymbolicConsequent
«definition»
OccurrenceComponent
DeductiveRulePart
ReactiveReductionPart


«definition»FunctionalRule
«exists» +part[1..*] : ReturnComponent
+returning {subsets taking}
«definition» ReturnComponent
«definition» RuleComponent
+as {subsets if, subsets returning}
«definition» ValueComponent
«definition» DerivationRule
{complete}
«definition» LogicalDerivationRule
+holds {subsets deriving}
«definition» ConsequentComponent
«definition» ReactiveDerivationRule
+occurrs {subsets deriving}
{complete}
«definition» ReactiveDeductiveRule
«exists» +part[1..*] : EventComponent
+when {subsets on, subsets returning}
«definition» EventComponent
«definition» ReactiveReductionRule
+part[*] : ReactiveReductionPart
«exists» +part[1..*] : ReducedOccurrence
«exists» +part[1..*] : ActionComponent
+to {subsets occurrs}
«definition» ReducedOccurrence
«definition» ReducedComponent
+is[1] : ReducedElement
«definition» OccurrenceComponent
«definition» ActionComponent
+must {subsets then, subsets returning}


Language
+in
LanguageItem
+in[0..1] : Language
+defines
{overlapping, complete}
+type
+type-is {subsets type, subsets sub-type}
LanguageFunctor
LanguageType
+type[*] : LanguageType
+sub-type
{complete}
SymbolicType
+sub-type[*] : SymbolicType
{complete}
LiteralType
+type[*] : LiteralType
«definition»
LexicalType
+uri[1] : xsd:anyURI
«definition»
DomainItem
+of[1] : Domain
Domain
+of {subsets in, subsets type}
+contains {subsets defines, subsets sub-type}


Algebra
+defines[*] : AlgebraicItem
+in
+defines
Domain
+of {subsets in, subsets type}
+contains {subsets defines, subsets sub-type}
«definition»
DomainItem
LanguageItem
AlgebraicItem
{overlapping, complete}
«definition»
AlgebraFunctor
+of[1] : Algebra
LanguageParameter
OperatorItem
LiteralType
«definition»
SymbolicFunctor
LanguageComponent
{complete}
«definition»
Component
«definition»
OperatorArgument
«definition»Operator
+item[*] : OperatorItem
«exists» +item[1..*] : OperatorArgument
{complete}
«definition»Argument
+is[1] : OperatorArgument
LanguageConstruct


FunctorItem
+default[0..1] : Constant
+builder[0..1] : LanguageFunctor
+default
«definition»
Constant
«definition»
RequiredItem
{complete}
OptionalItem
+builder
LogicalParameter
{overlapping, complete}
«definition»
Term
GroundTerm
{complete}
LiteralValue
LanguageFunctor
«definition»
DefaultedItem
+default[1]
«definition»
StructuredItem
+builder[1]
«definition»
GroundComplexTerm
+part[*] : GroundPart
-with[0]
ComplexTerm
+nil {subsets default}
«definition»
CollectionItem
+collector[1] : CollectionFunctor
+collects-to {subsets item, subsets type-is}
«definition»
GroundPart
+bound-to[*] : GroundTerm
+equals[*] : GroundTerm
TermPart
+collector {subsets builder}
CollectionFunctor
+collects-to[1] : CollectionItem
«owl 1.1» +item[1] : RequiredItem


«definition»
FormulaConstruct
LogicalConstructItem
{complete}
FunctionalParameter
LanguageComponent
AntecedentElement
+default[0..1] : TrueConstant
+builder[0..1] : ConjunctionConstruct
LogicalConstruct
+item[*] : LogicalConstructItem
TrueConstant
«definition»
NilConstruct
Quantifier
«definition»UnaryConstruct
«owl 1.1» +item[1] : LanguageComponent
AssertConstruct
RetractConstruct
NegationConstruct
ExplicitNegationConstruct
DefaultNegationConstruct
NegationAsFailureConstruct
ConjunctionConstruct
«definition»BinaryConstruct
«owl 1.1» +item[2] : LanguageComponent
DisjunctionConstruct
EqualityConstruct
UnificationConstruct
StrictEqualityConstruct
AssignmentConstruct