OOA of MASL

xtuml.org

cortland.starrett@onefact.net

OneFact

ONE FACT, INC.

OOA of MASL

outline

- goals
- strategy
- OOA of MASL
- MASL loader
- next steps

goals

- capture MASL as a model
- parse and load correct MASL
- anticipate cloud architecture

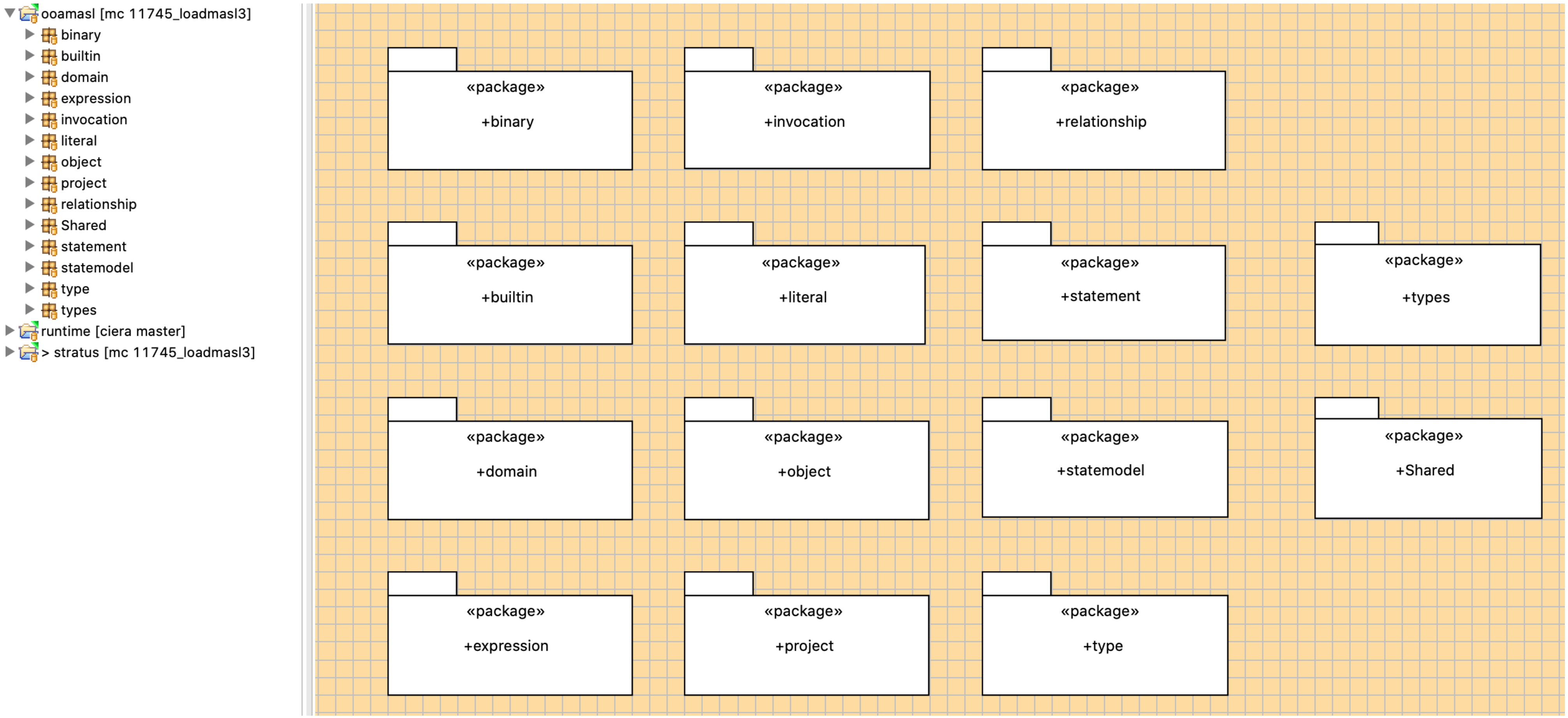
strategy

- port MASL model compiler java model
  - leverage working starting point
  - low-risk with large number of artifacts
  - script conversion of java to MASL
  - import into BridgePoint
- adjust ported model
- refine with action language

OOA of MASL

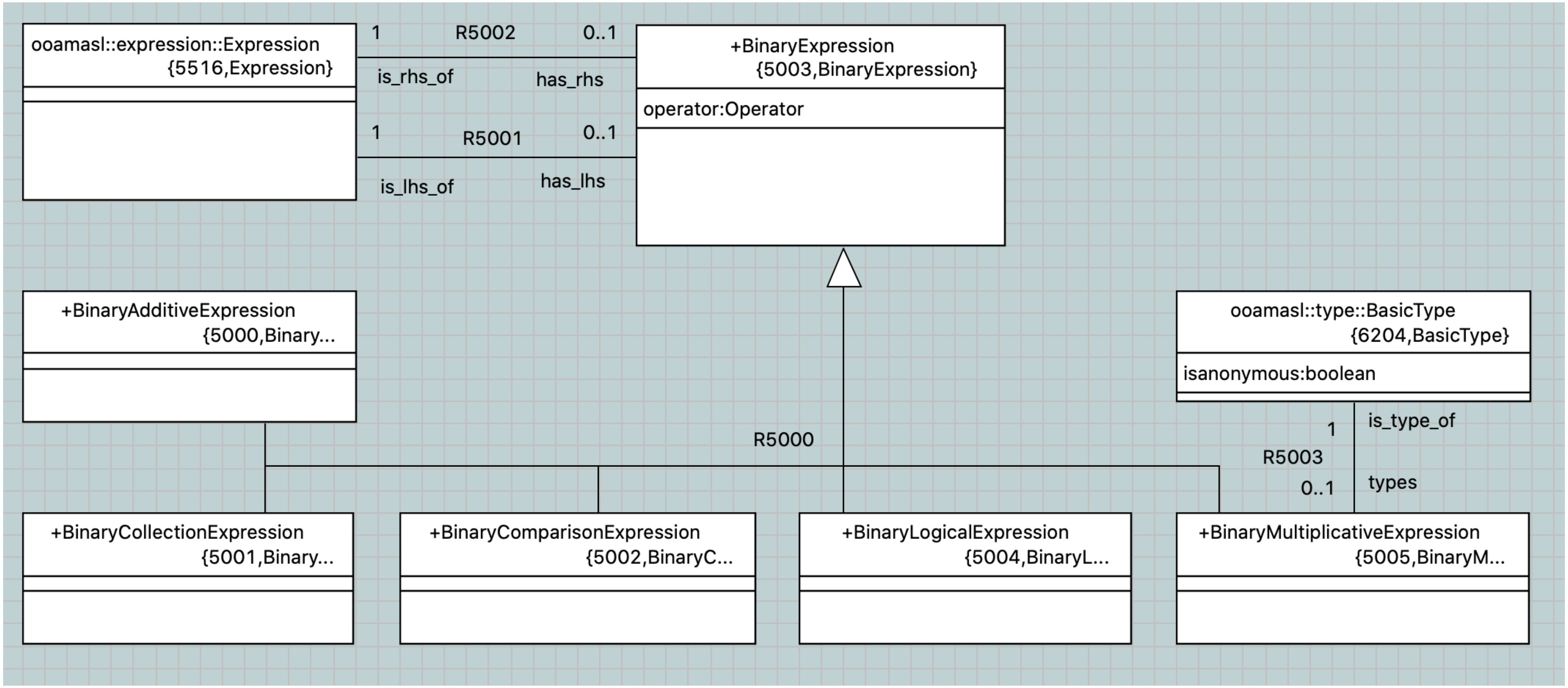
- ooamasl subsystem packages
- package references and *naked* OOA
- statistics
- deployment component (ME and canvas)
- tour of each subsystem

OOA of MASL: **ooamasl packages**

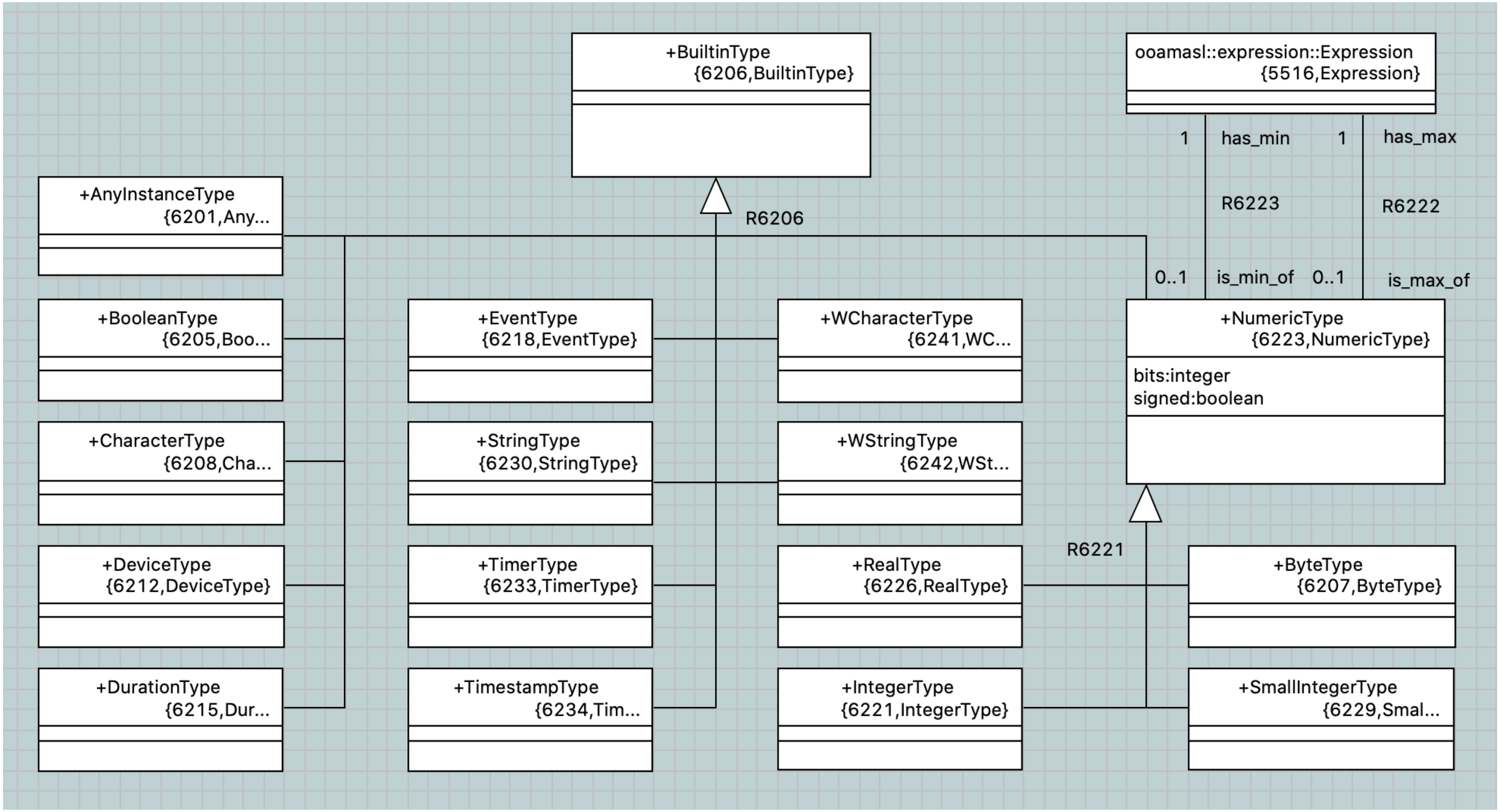


- ooamasl [mc 11745_loadmasl3]
  - binary
  - builtin
  - domain
  - expression
  - invocation
  - literal
  - object
  - project
  - relationship
  - Shared
  - statement
  - statemodel
  - type
  - types
- runtime [ciera master]
- > stratus [mc 11745_loadmasl3]

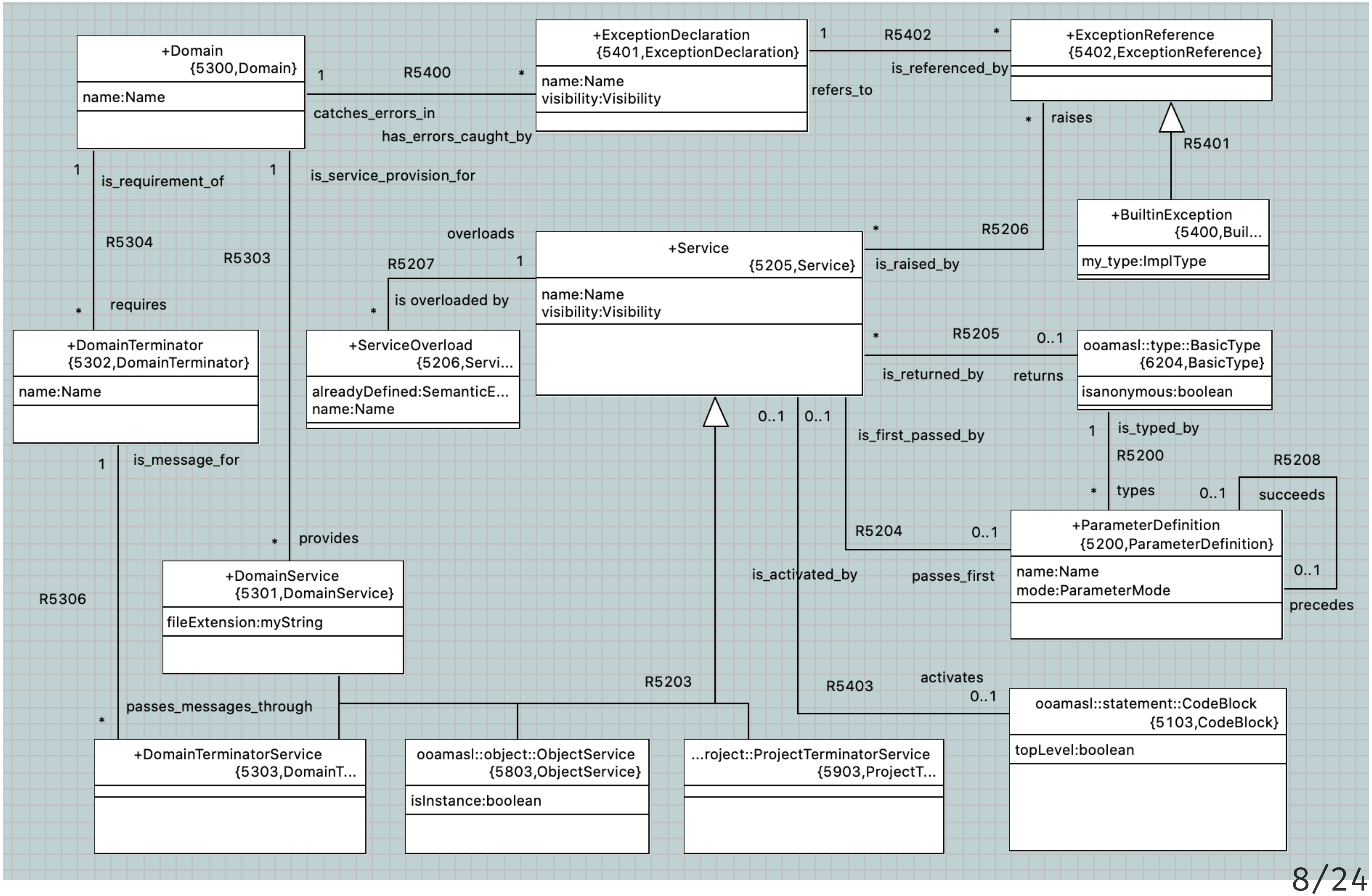
OOA of MASL: **binary**



OOA of MASL: **builtin**



OOA of MASL: **domain**



+Domain
{5300,Domain}
name:Name

+ExceptionDeclaration
{5401,ExceptionDeclaration}
name:Name
visibility:Visibility

+ExceptionReference
{5402,ExceptionReference}

+BuiltinException
{5400,Buil...
my_type:ImplType

+Service
{5205,Service}
name:Name
visibility:Visibility

+DomainTerminator
{5302,DomainTerminator}
name:Name

+ServiceOverload
{5206,Servi...
alreadyDefined:SemanticE...
name:Name

ooamasl::type::BasicType
{6204,BasicType}
isanonymous:boolean

+ParameterDefinition
{5200,ParameterDefinition}
name:Name
mode:ParameterMode

+DomainService
{5301,DomainService}
fileExtension:myString

ooamasl::statement::CodeBlock
{5103,CodeBlock}
topLevel:boolean

+DomainTerminatorService
{5303,DomainT...

ooamasl::object::ObjectService
{5803,ObjectService}
isInstance:boolean

...roject::ProjectTerminatorService
{5903,ProjectT...

1 R5400 *
catches_errors_in
has_errors_caught_by

1 R5402 *
is_referenced_by
refers_to

R5401

* raises
R5206
* is_raised_by

1 is_requirement_of
R5304
* requires

1 is_service_provision_for
R5303
* provides

overloads
R5207 1
* is overloaded by

* R5205 0..1
is_returned_by returns

0..1 0..1
is_first_passed_by
R5204 0..1
passes_first

1 is_typed_by
R5200
* types

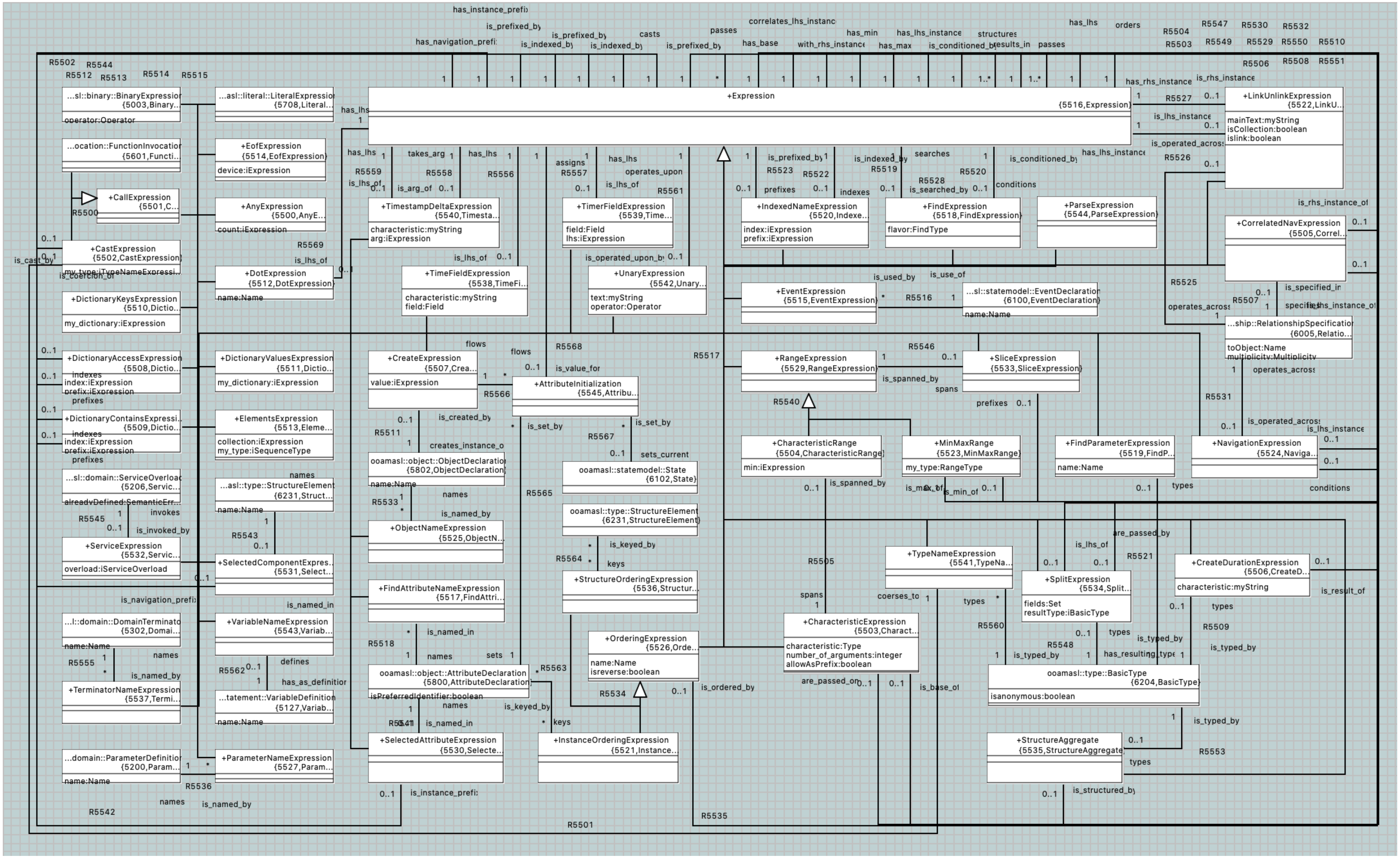
R5208
0..1 succeeds
0..1 precedes

1 is_message_for
R5306
* passes_messages_through

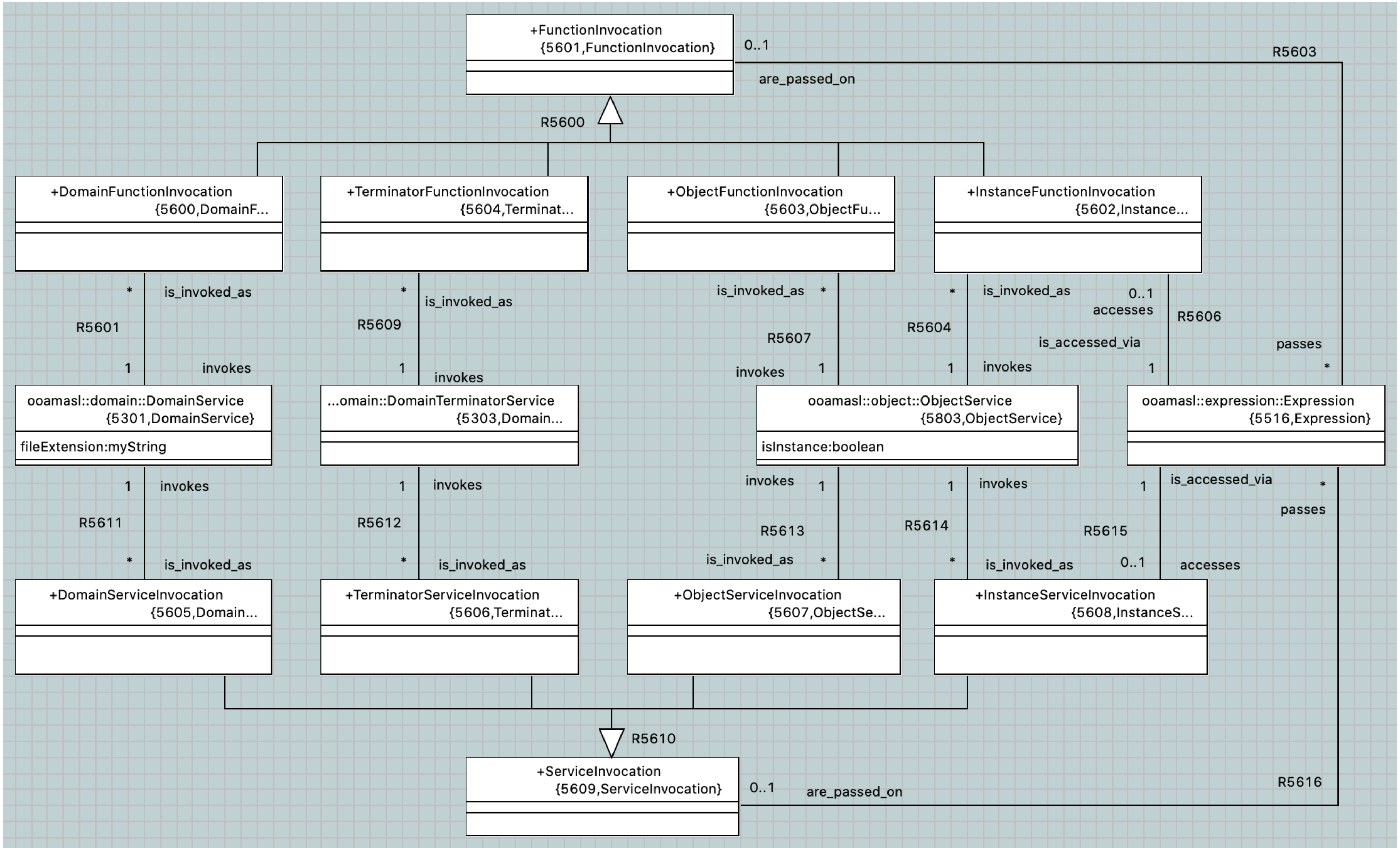
R5203

is_activated_by
R5403
activates
0..1

OOA of MASL: **expression**



OOA of MASL: **invocation**



+FunctionInvocation
{5601,FunctionInvocation}

0..1

R5603

are_passed_on

R5600

+DomainFunctionInvocation
{5600,DomainF...

+TerminatorFunctionInvocation
{5604,Terminat...

+ObjectFunctionInvocation
{5603,ObjectFu...

+InstanceFunctionInvocation
{5602,Instance...

*
is_invoked_as
R5601
1
invokes

*
is_invoked_as
R5609
1
invokes

is_invoked_as
*
R5607
invokes
1

*
is_invoked_as
R5604
1
invokes

0..1
accesses
R5606
is_accessed_via
1

passes
*

ooamasl::domain::DomainService
{5301,DomainService}
fileExtension:myString

...omain::DomainTerminatorService
{5303,Domain...

ooamasl::object::ObjectService
{5803,ObjectService}
isInstance:boolean

ooamasl::expression::Expression
{5516,Expression}

1
invokes
R5611
*
is_invoked_as

1
invokes
R5612
*
is_invoked_as

invokes
1
R5613
is_invoked_as
*

1
invokes
R5614
*
is_invoked_as

1
is_accessed_via
R5615
0..1
accesses

*
passes

+DomainServiceInvocation
{5605,Domain...

+TerminatorServiceInvocation
{5606,Terminat...

+ObjectServiceInvocation
{5607,ObjectSe...

+InstanceServiceInvocation
{5608,InstanceS...

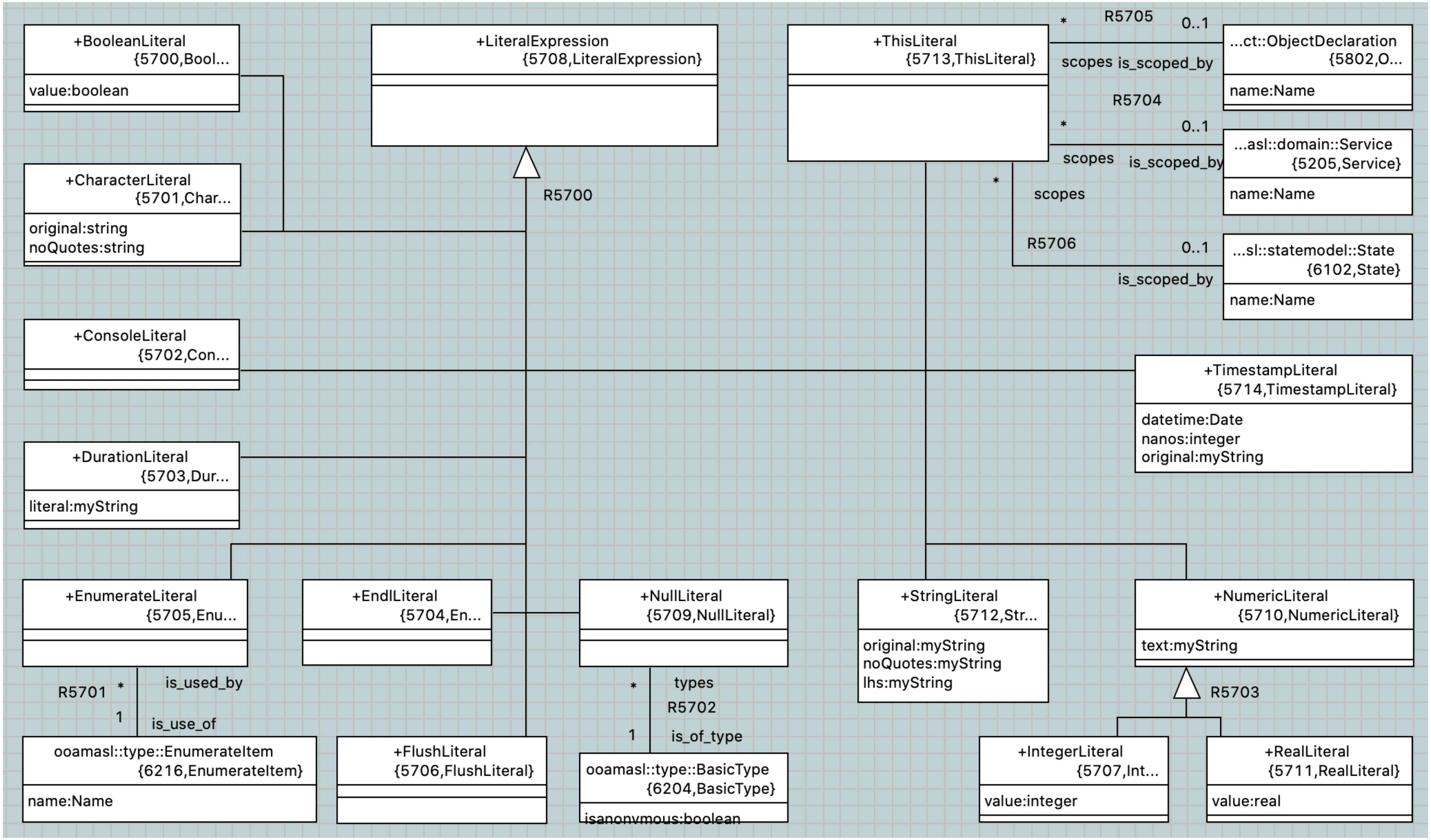
R5610

+ServiceInvocation
{5609,ServiceInvocation}

0..1
are_passed_on

R5616

OOA of MASL: **literal**



+BooleanLiteral
{5700,Bool...
value:boolean

+LiteralExpression
{5708,LiteralExpression}

+ThisLiteral
{5713,ThisLiteral}

* R5705 0..1
scopes is_scoped_by

...ct::ObjectDeclaration
{5802,O...
name:Name

R5704
* 0..1
scopes is_scoped_by

...asl::domain::Service
{5205,Service}
name:Name

R5700

+CharacterLiteral
{5701,Char...
original:string
noQuotes:string

* scopes
R5706 0..1
is_scoped_by

...sl::statemodel::State
{6102,State}
name:Name

+ConsoleLiteral
{5702,Con...

+TimestampLiteral
{5714,TimestampLiteral}
datetime:Date
nanos:integer
original:myString

+DurationLiteral
{5703,Dur...
literal:myString

+EnumerateLiteral
{5705,Enu...

+EndlLiteral
{5704,En...

+NullLiteral
{5709,NullLiteral}

+StringLiteral
{5712,Str...
original:myString
noQuotes:myString
lhs:myString

+NumericLiteral
{5710,NumericLiteral}
text:myString

R5701 * is_used_by
1 is_use_of

* types
R5702
1 is_of_type

R5703

ooamasl::type::EnumerateItem
{6216,EnumerateItem}
name:Name

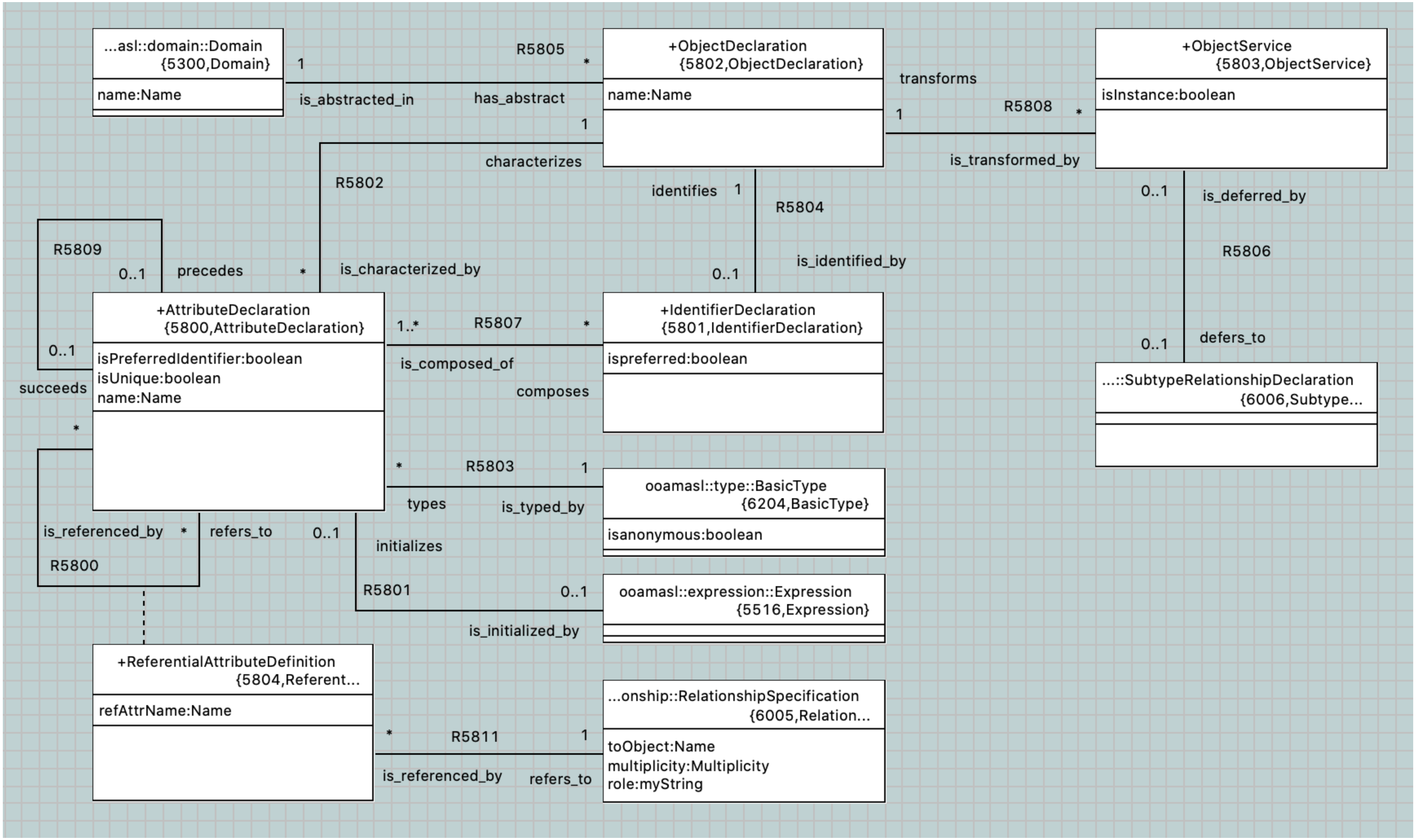
+FlushLiteral
{5706,FlushLiteral}

ooamasl::type::BasicType
{6204,BasicType}
isanonymous:boolean

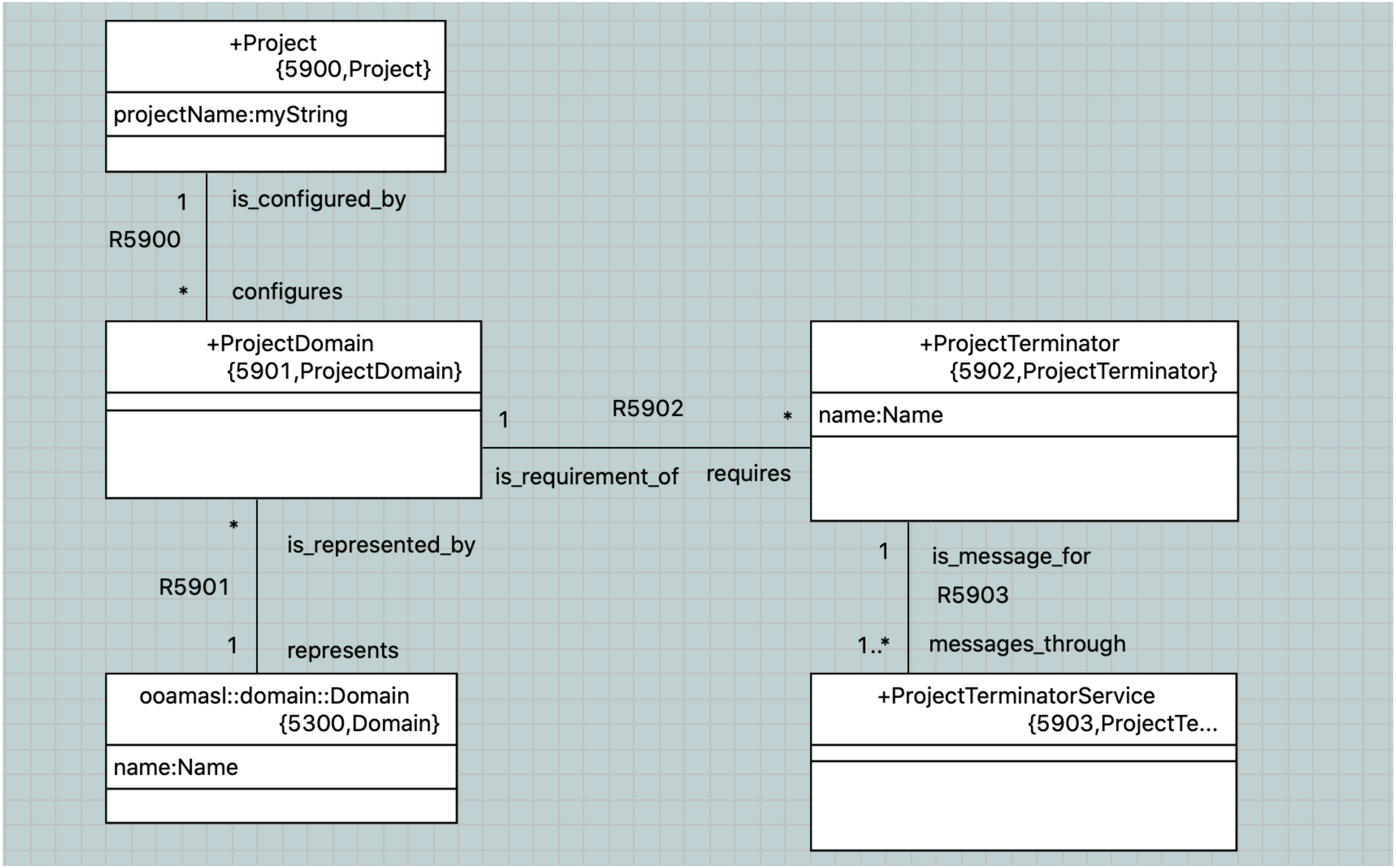
+IntegerLiteral
{5707,Int...
value:integer

+RealLiteral
{5711,RealLiteral}
value:real

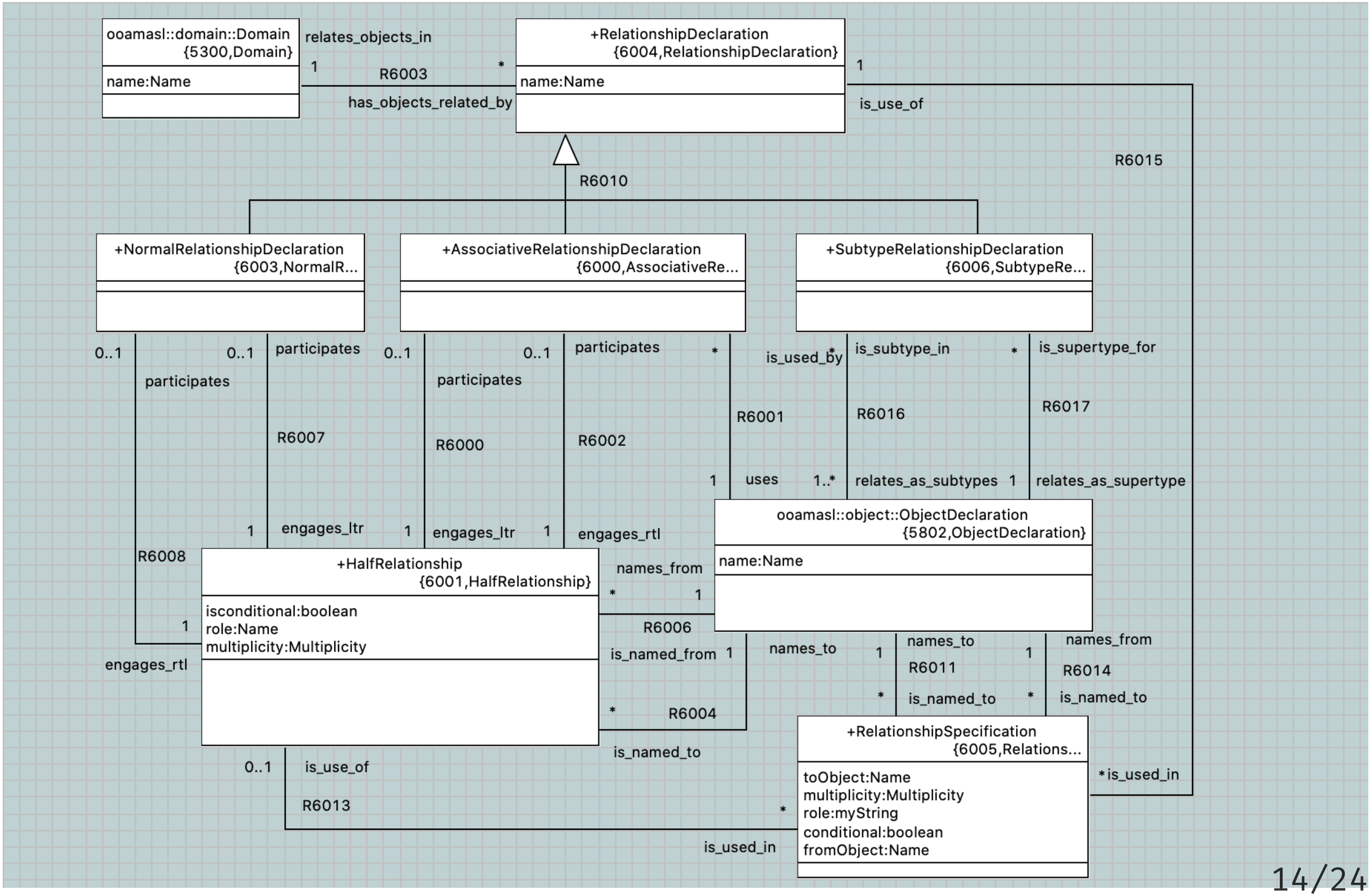
OOA of MASL: **object**



OOA of MASL: **project**



OOA of MASL: relationship



ooamasl::domain::Domain
{5300,Domain}
name:Name

relates_objects_in
1
R6003
*
has_objects_related_by

+RelationshipDeclaration
{6004,RelationshipDeclaration}
name:Name

1
is_use_of
R6015

R6010

+NormalRelationshipDeclaration
{6003,NormalR...

+AssociativeRelationshipDeclaration
{6000,AssociativeRe...

+SubtypeRelationshipDeclaration
{6006,SubtypeRe...

0..1
participates
0..1
participates
R6007
R6008
1
engages_ltr
1
engages_rtl

0..1
participates
0..1
participates
R6000
R6002
1
engages_ltr
1
engages_rtl

*
is_used_by
R6001
1
uses

*
is_subtype_in
R6016
1..*
relates_as_subtypes

*
is_supertype_for
R6017
1
relates_as_supertype

ooamasl::object::ObjectDeclaration
{5802,ObjectDeclaration}
name:Name

+HalfRelationship
{6001,HalfRelationship}
isconditional:boolean
role:Name
multiplicity:Multiplicity

names_from
*
1
R6006
is_named_from
1
names_to
*
R6004
is_named_to

1
names_to
R6011
*
is_named_to

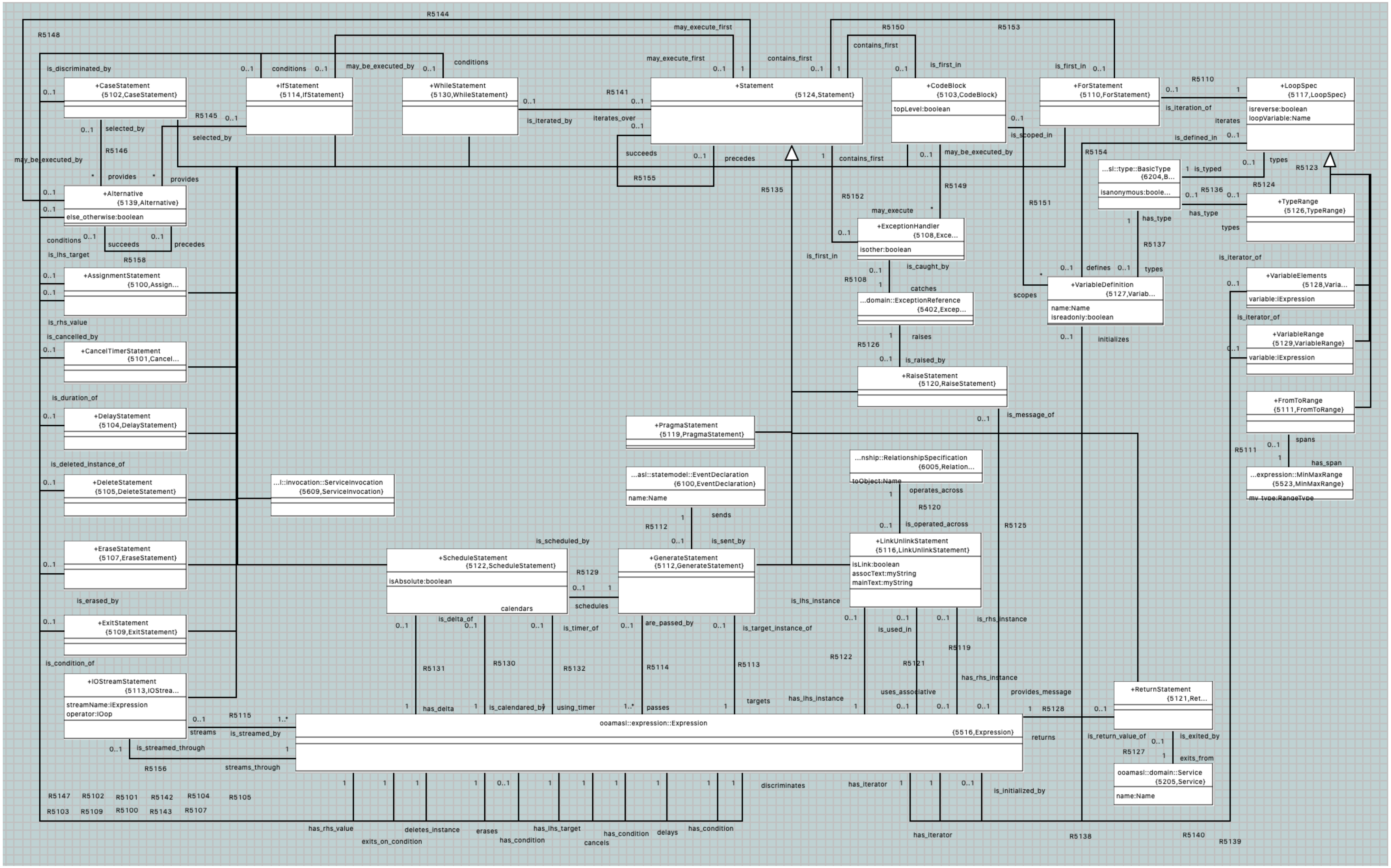
1
names_from
R6014
*
is_named_to

0..1
is_use_of
R6013
*
is_used_in

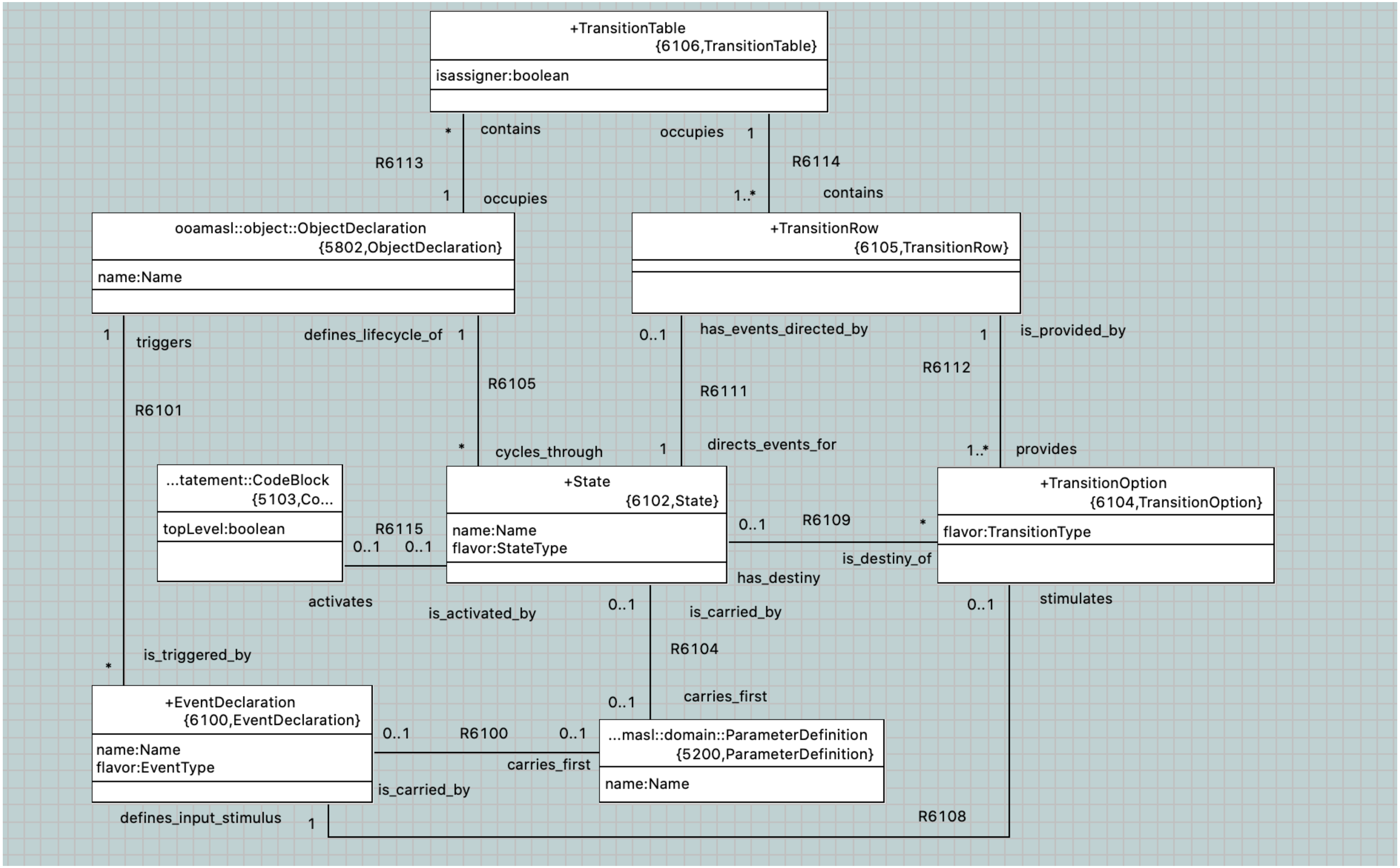
+RelationshipSpecification
{6005,Relations...
toObject:Name
multiplicity:Multiplicity
role:myString
conditional:boolean
fromObject:Name

*is_used_in

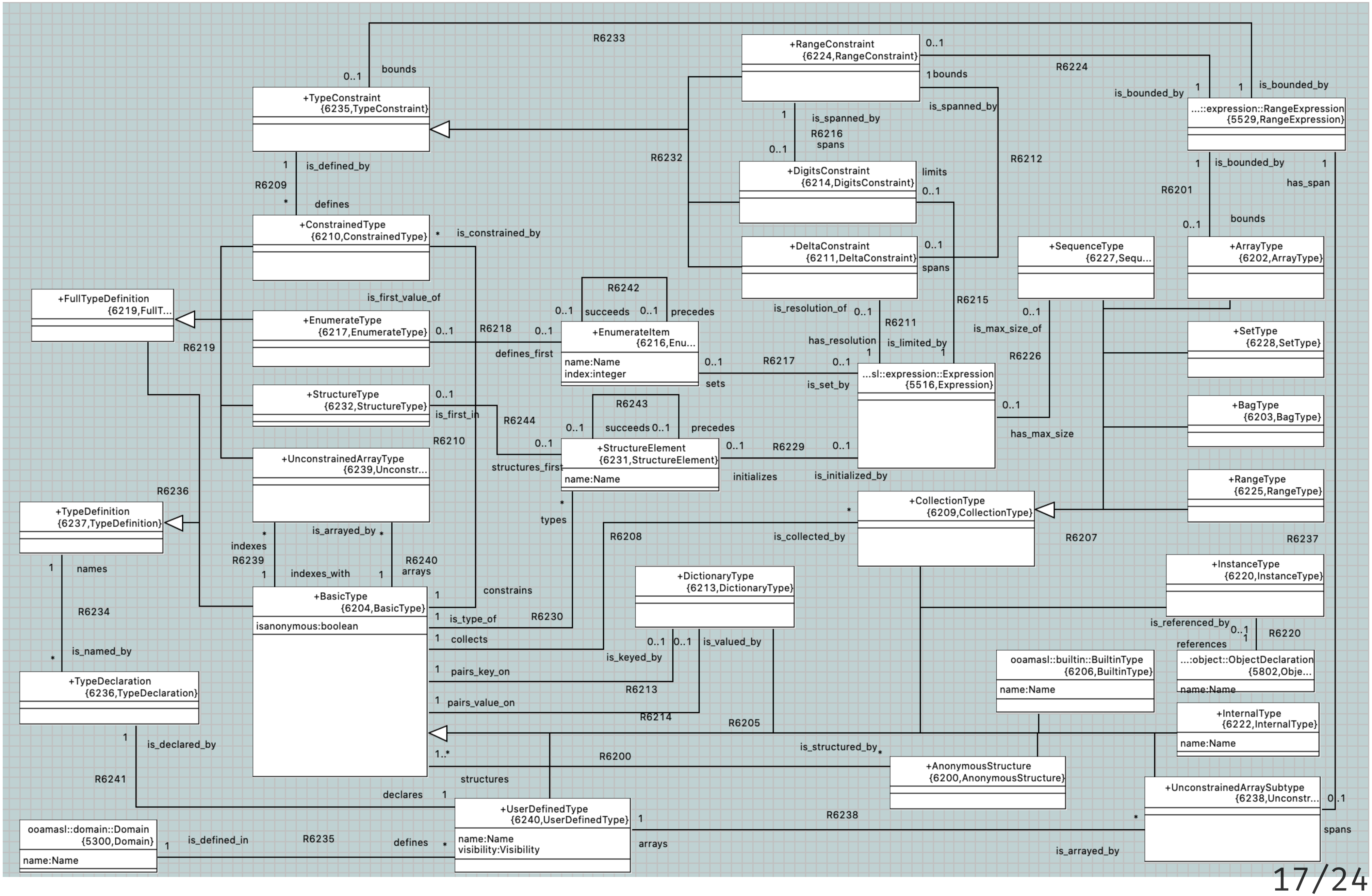
OOA of MASL: statement



OOA of MASL: statemodel



OOA of MASL: **type**



package references and *naked* OOA of MASL

- A package reference is a package linked to another package in another location (package or project).
    - Package references allow reuse and *import* of subsystems.
- The term *naked OOA* refers to a class model with no state or functional activity intended to be used as the schema for multiple projects.
    - A meta-model is most useful as a naked class model.

Statistics

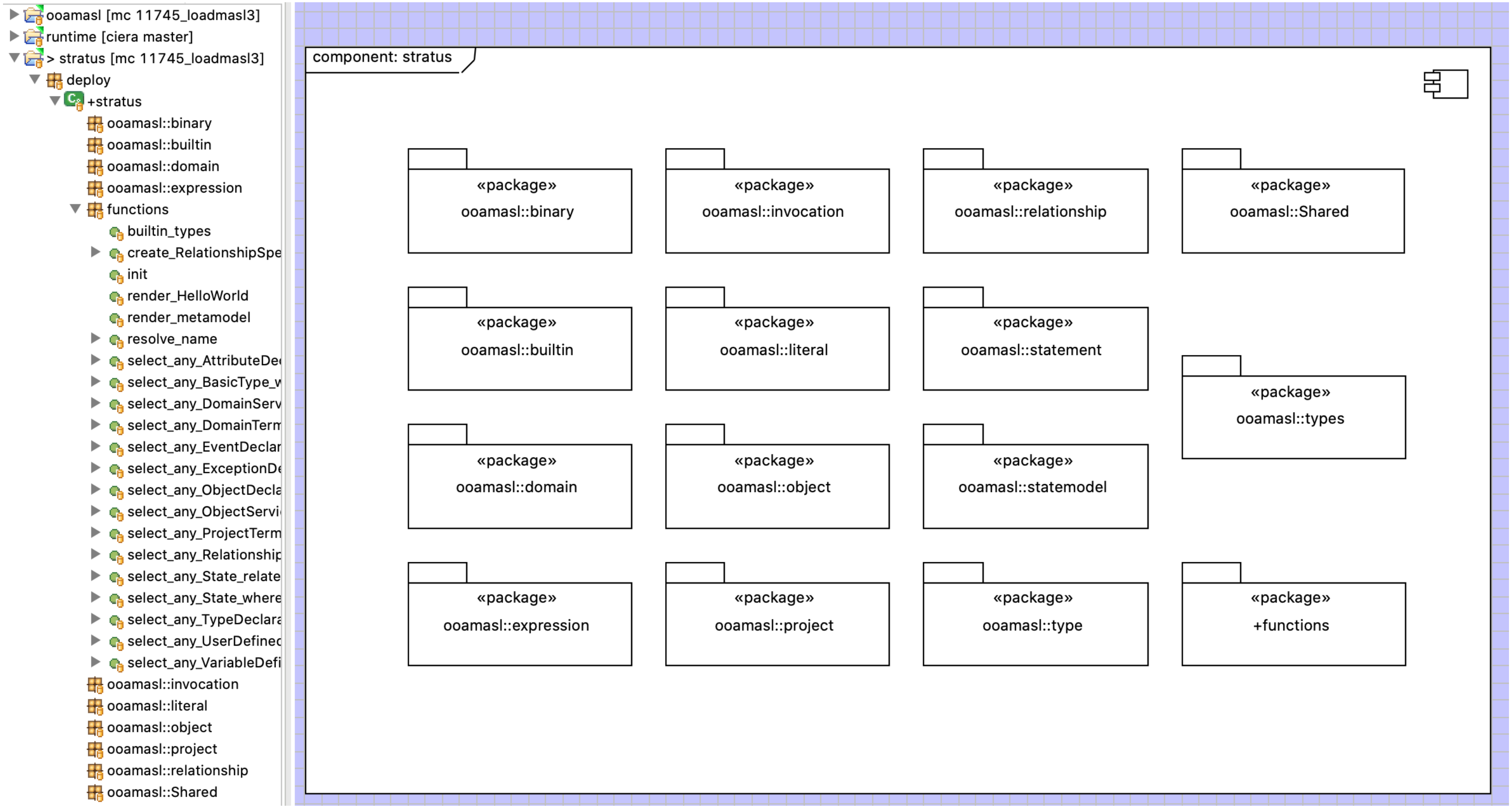
Statistics

subsystems	12
classes	176
attributes	129
relationships	230
production rules	166

MASL loader

- deployment model
  - imported OOA of MASL subsystem packages
  - loader helper functions (OAL)
- parser as `antlr` lexer / parser
- loader as `antlr` walker with invocations into deployment model
- model API functions
  - Ciera provides an API which allows `create`, `relate`, `set_attribute` and `call_function`

OOA of MASL: Stratus deployment



```
▶ ooamasl [mc 11745_loadmasl3]
▶ runtime [ciera master]
▼ > stratus [mc 11745_loadmasl3]
    ▼ deploy
        ▼ +stratus
              ooamasl::binary
              ooamasl::builtin
              ooamasl::domain
              ooamasl::expression
            ▼ functions
                  builtin_types
                ▶ create_RelationshipSpe
                  init
                  render_HelloWorld
                  render_metamodel
                ▶ resolve_name
                ▶ select_any_AttributeDe
                ▶ select_any_BasicType_w
                ▶ select_any_DomainServ
                ▶ select_any_DomainTerm
                ▶ select_any_EventDeclar
                ▶ select_any_ExceptionDe
                ▶ select_any_ObjectDecla
                ▶ select_any_ObjectServi
                ▶ select_any_ProjectTerm
                ▶ select_any_Relationship
                ▶ select_any_State_relate
                ▶ select_any_State_where
                ▶ select_any_TypeDeclara
                ▶ select_any_UserDefinec
                ▶ select_any_VariableDefi
              ooamasl::invocation
              ooamasl::literal
              ooamasl::object
              ooamasl::project
              ooamasl::relationship
              ooamasl::Shared
```

component: stratus

«package» ooamasl::binary	«package» ooamasl::invocation	«package» ooamasl::relationship	«package» ooamasl::Shared
«package» ooamasl::builtin	«package» ooamasl::literal	«package» ooamasl::statement	«package» ooamasl::types
«package» ooamasl::domain	«package» ooamasl::object	«package» ooamasl::statemodel	
«package» ooamasl::expression	«package» ooamasl::project	«package» ooamasl::type	«package» +functions

MASL loader

- action language functions
- `LOAD` class interface to generated model Java
- Ciera provides a build chain based upon `maven`.
- Ciera generated code
- calculator ALU has been primary test model.
- Stratus can parse and load Stratus.
- Generate Hello, world.
- Generate abstract classes with stratus of stratus.
- Generate inventory report of major elements sorted in some order.

MASL loader: key functions

- init
- Builtin/InternalType_populate
- select_any_ObjectDeclaration_where_name
- resolve_name
- create_RelationshipSpecification
- ReferentialAttribute_resolve

next steps

- review, further test and refinement
- template and rendering HOWTO
- cloud template integration
- Java translation of MASL activities