OOA of MASL

xtuml.org

cortland.starrett@onefact.net

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

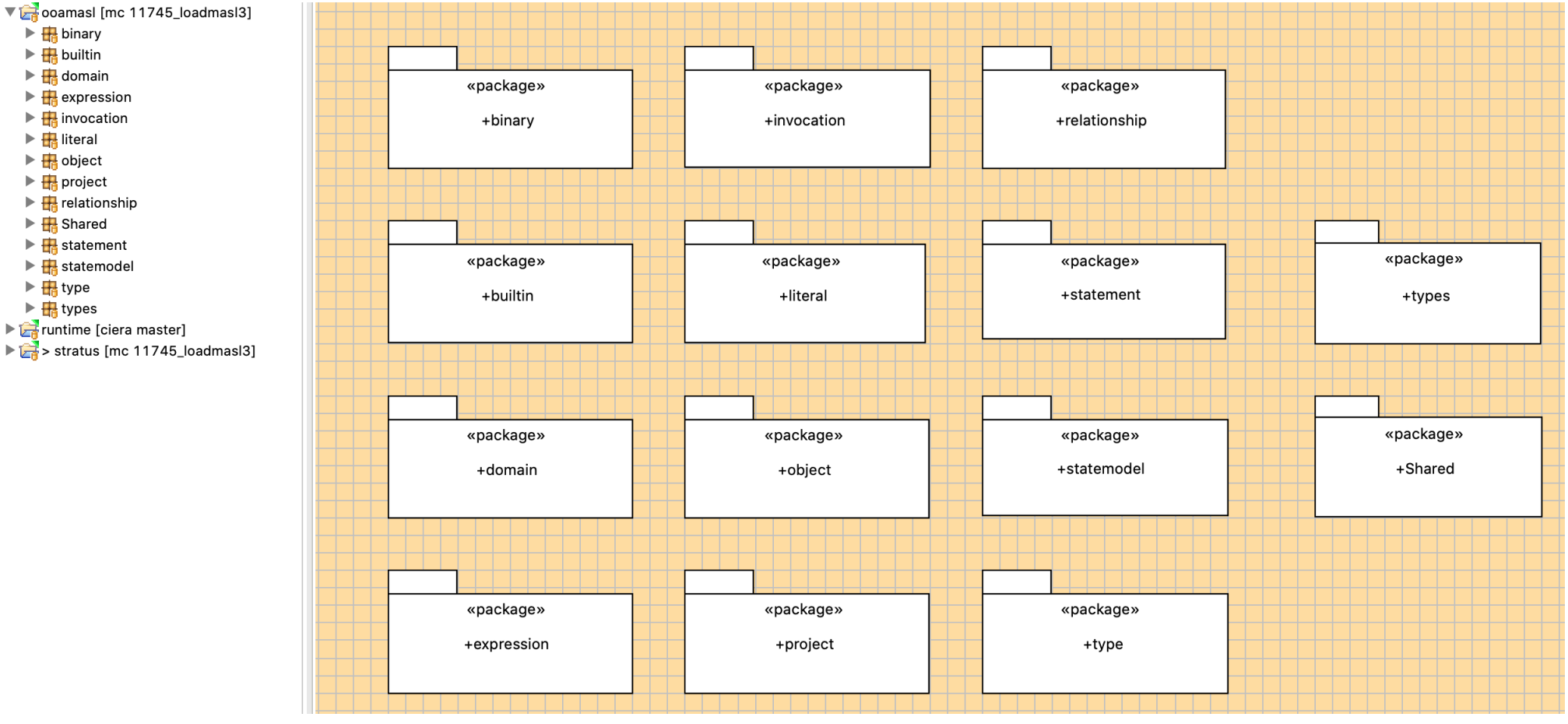
strategy

- 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

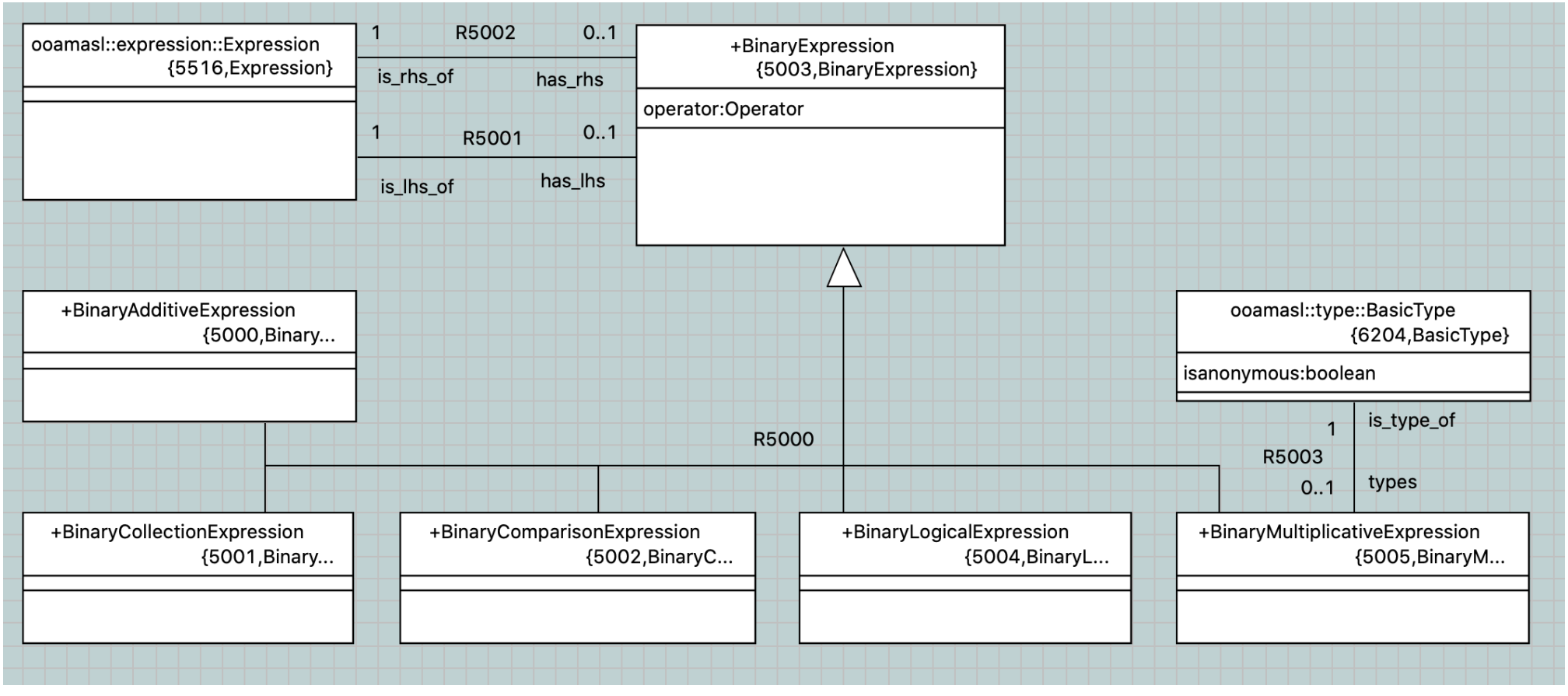
OOA of MASL

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

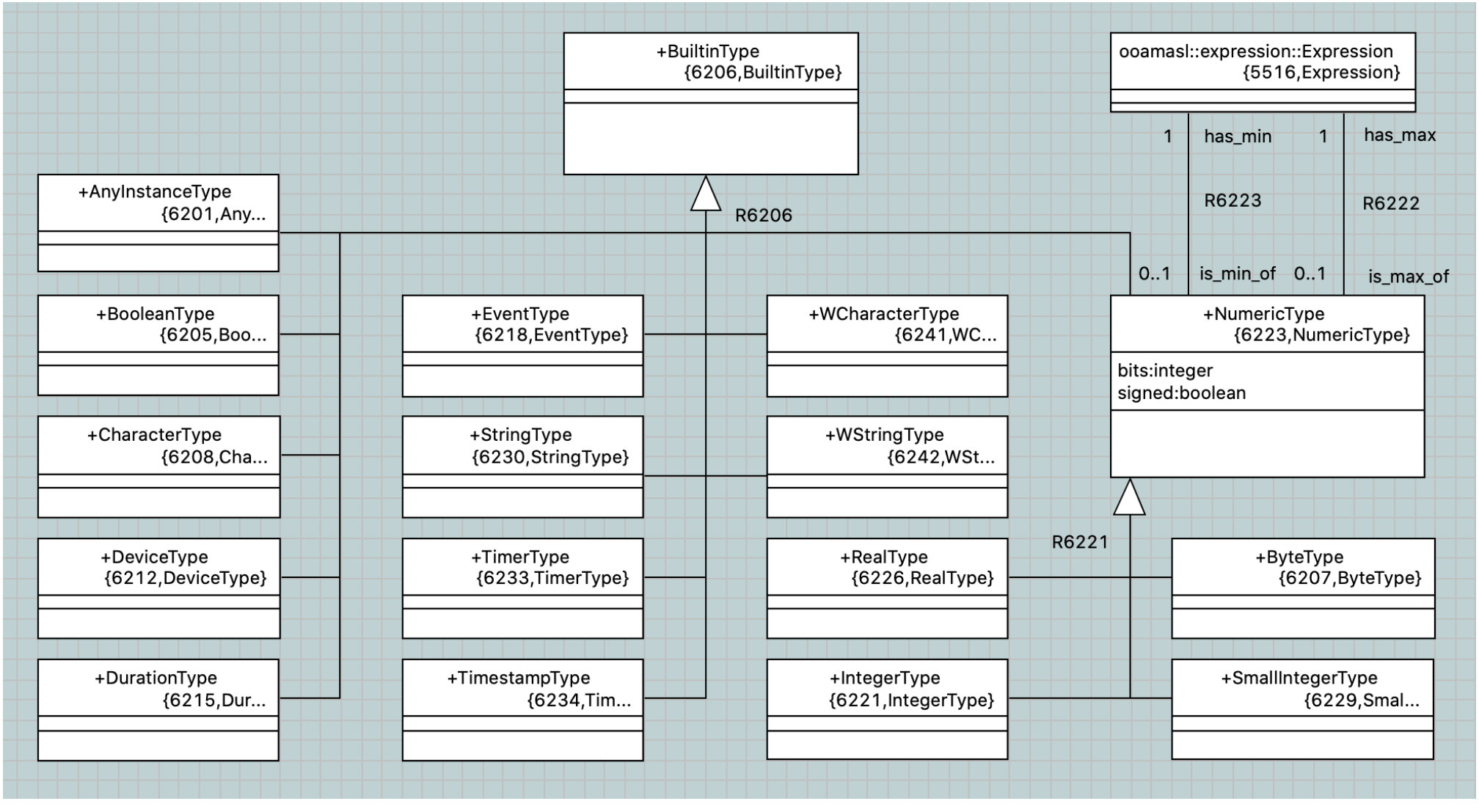
OOA of MASL: ooamasl packages



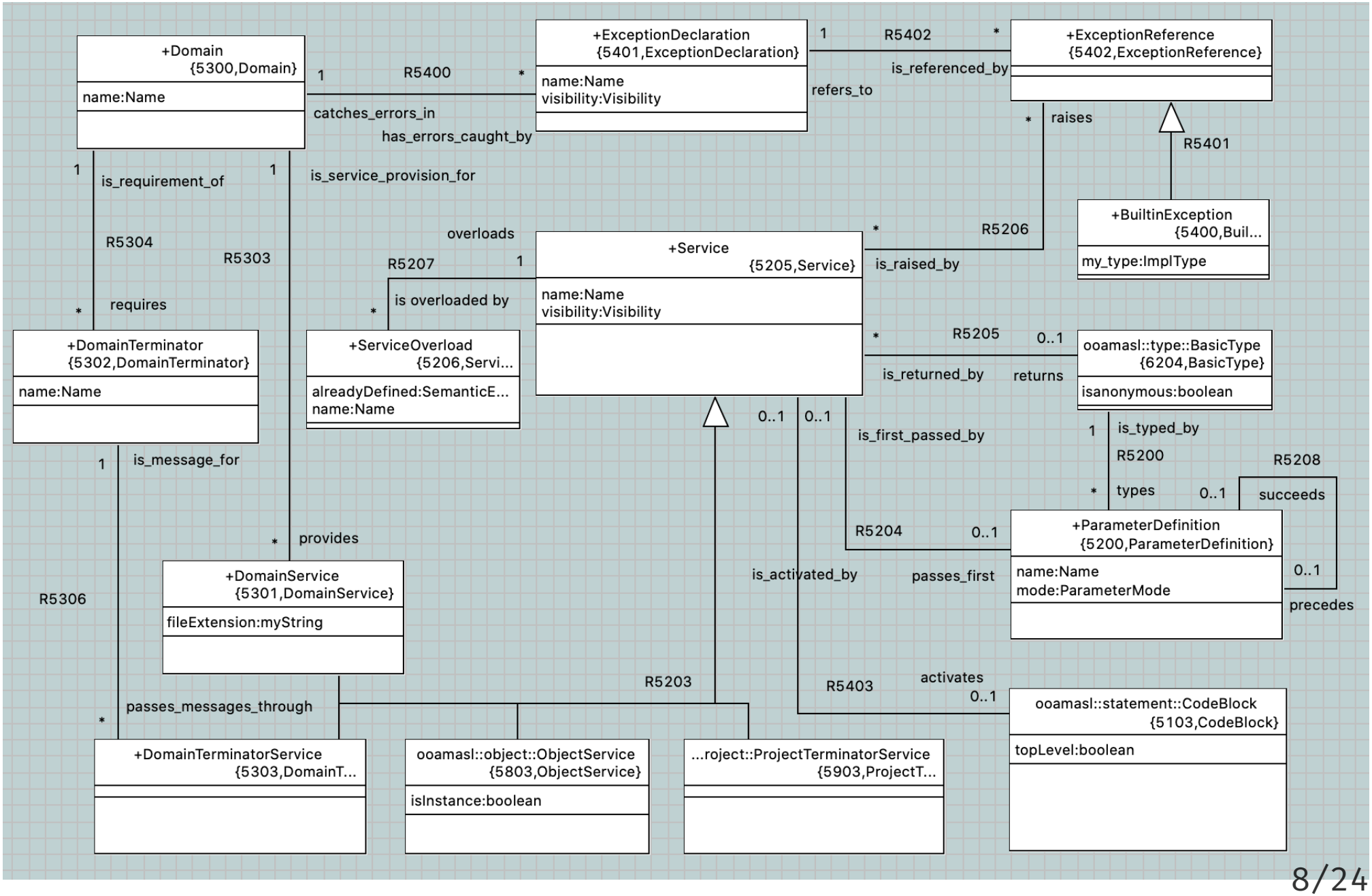
OOA of MASL: binary



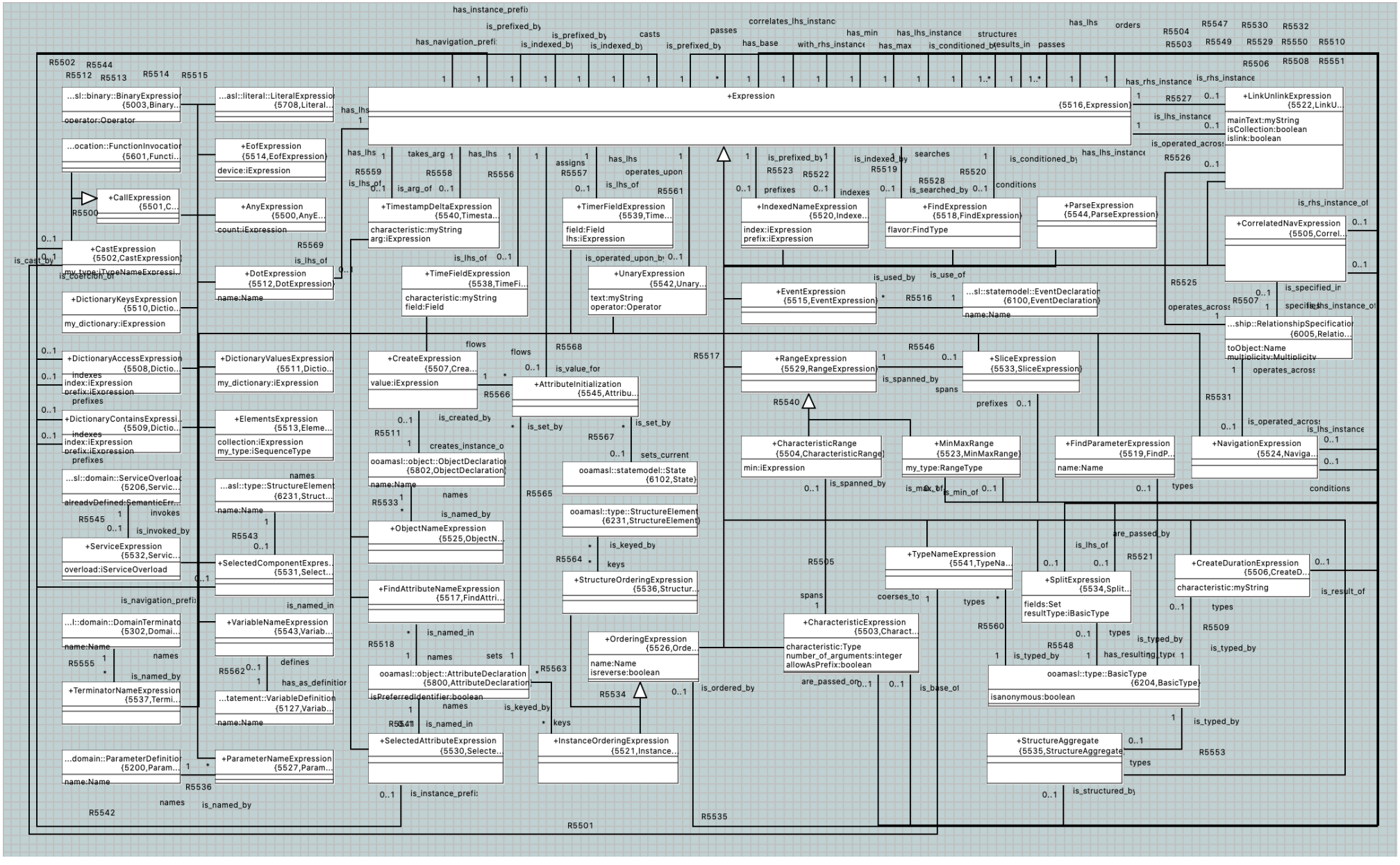
OOA of MASL: builtin



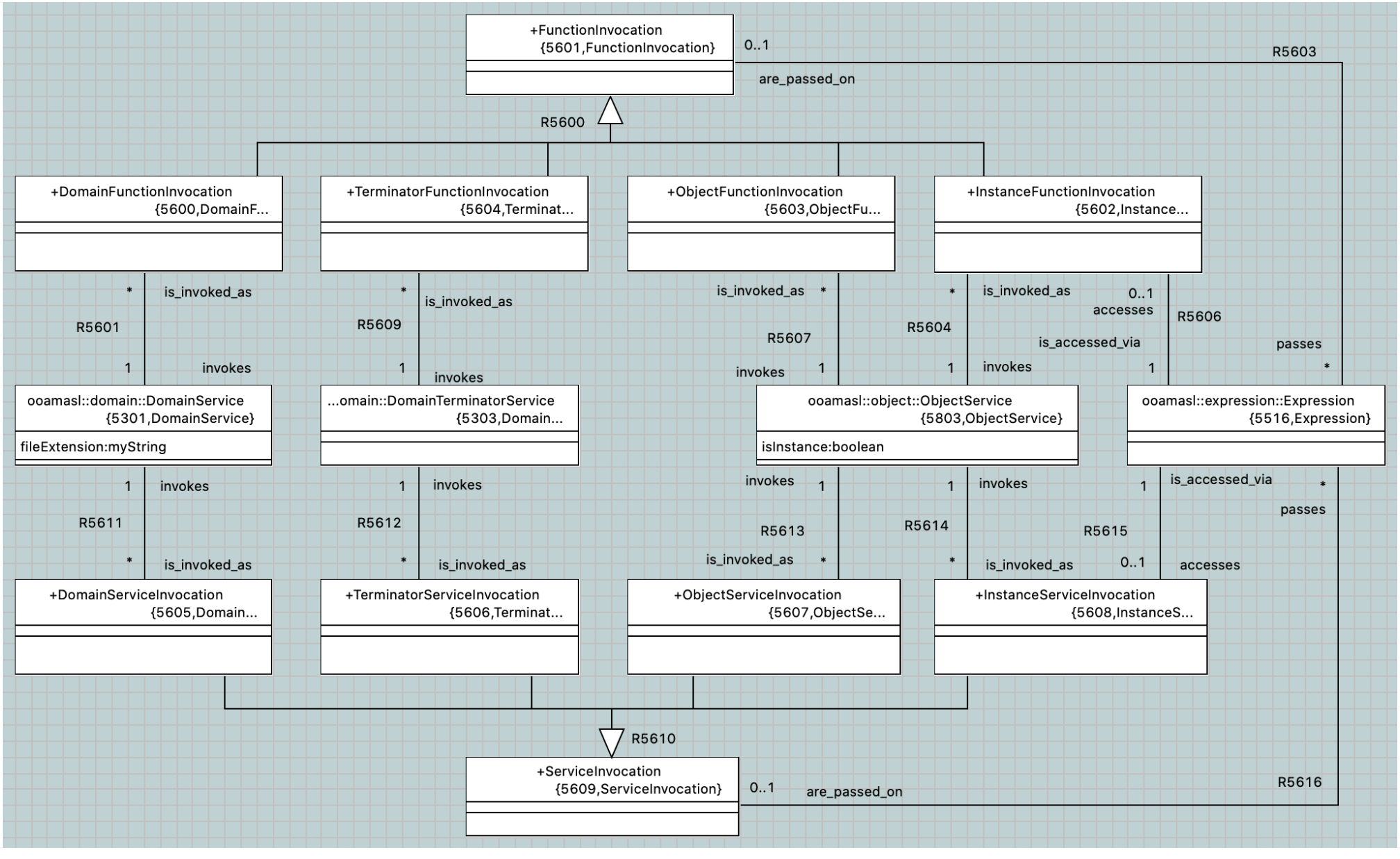
OOA of MASL: domain



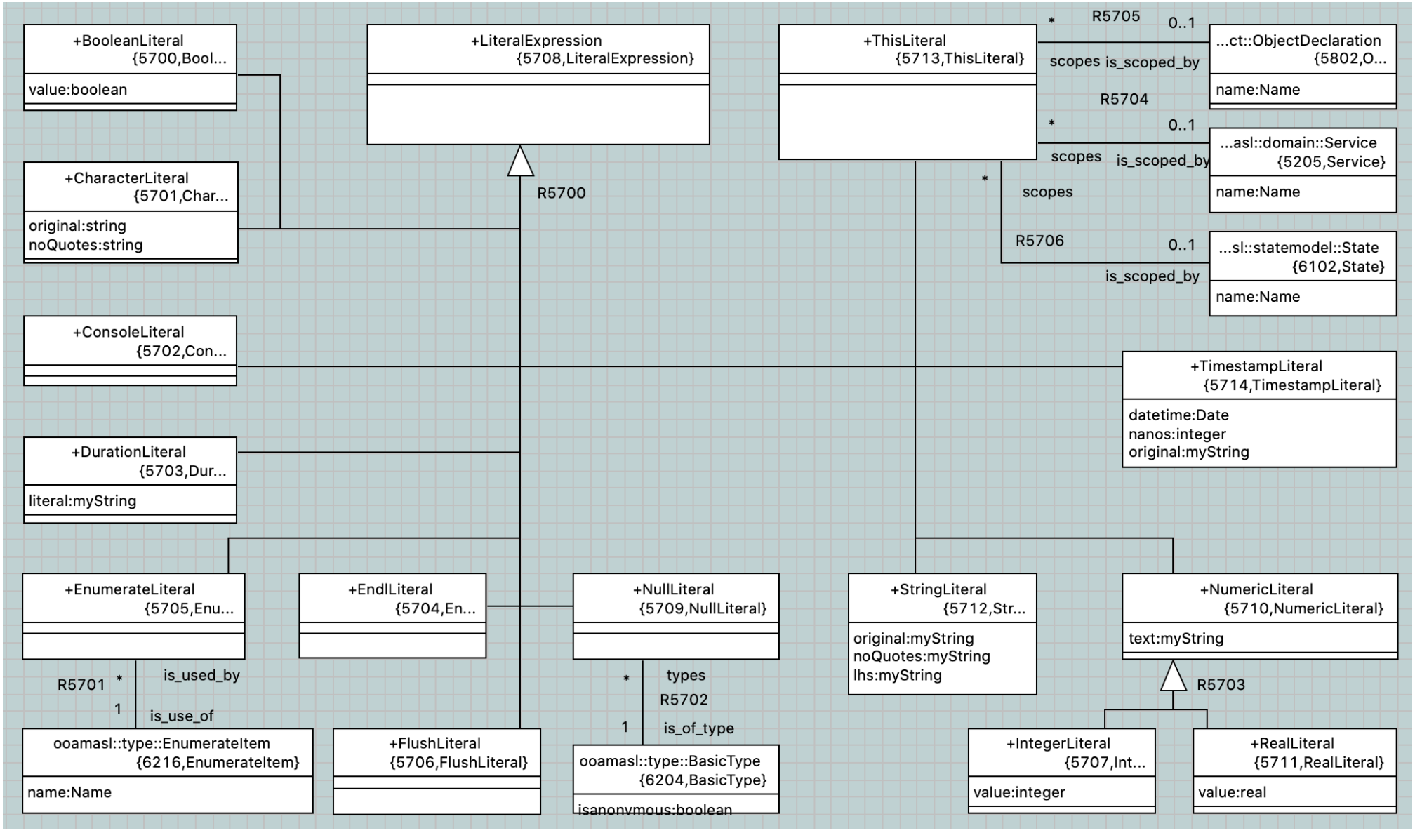
OOA of MASL: expression



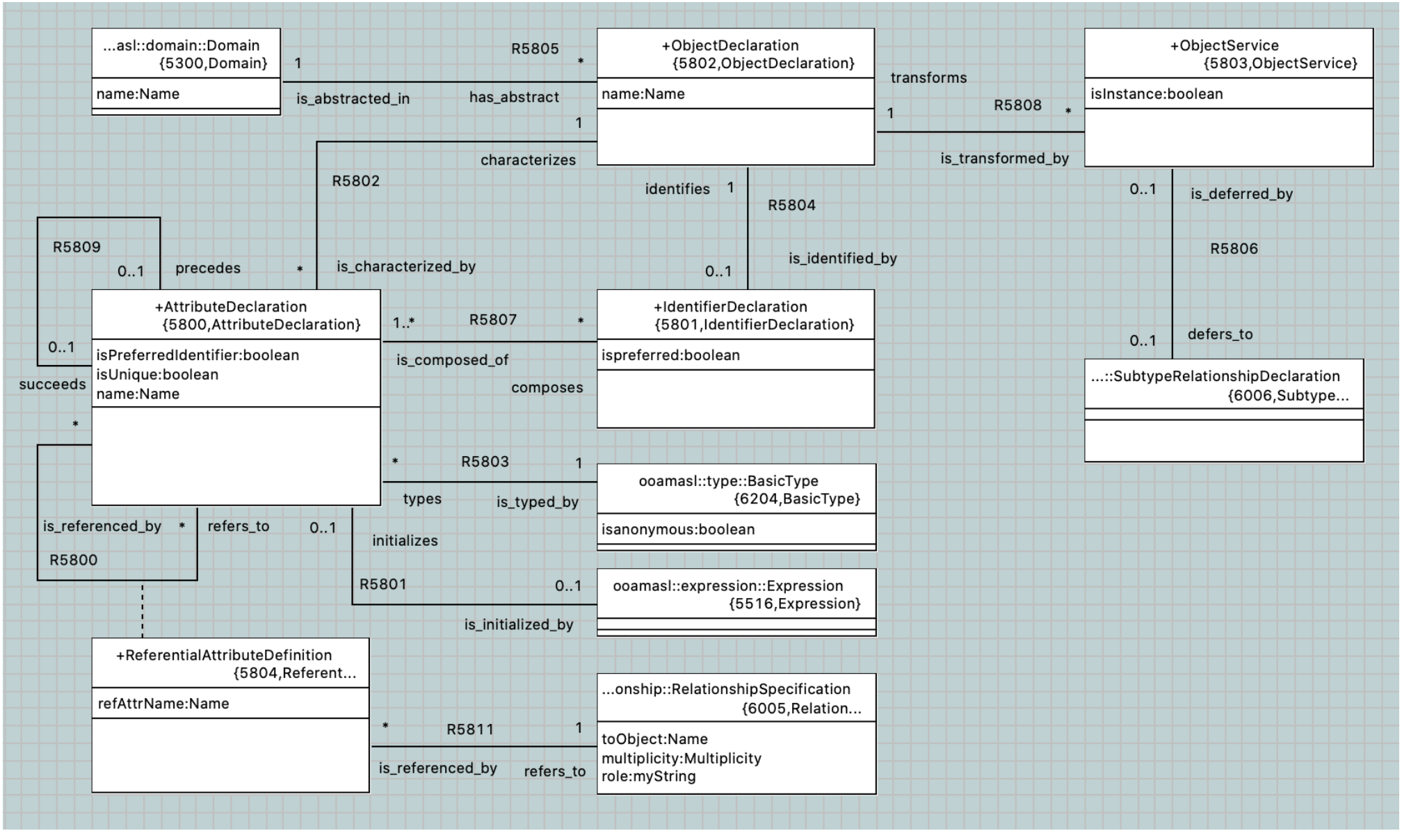
OOA of MASL: invocation



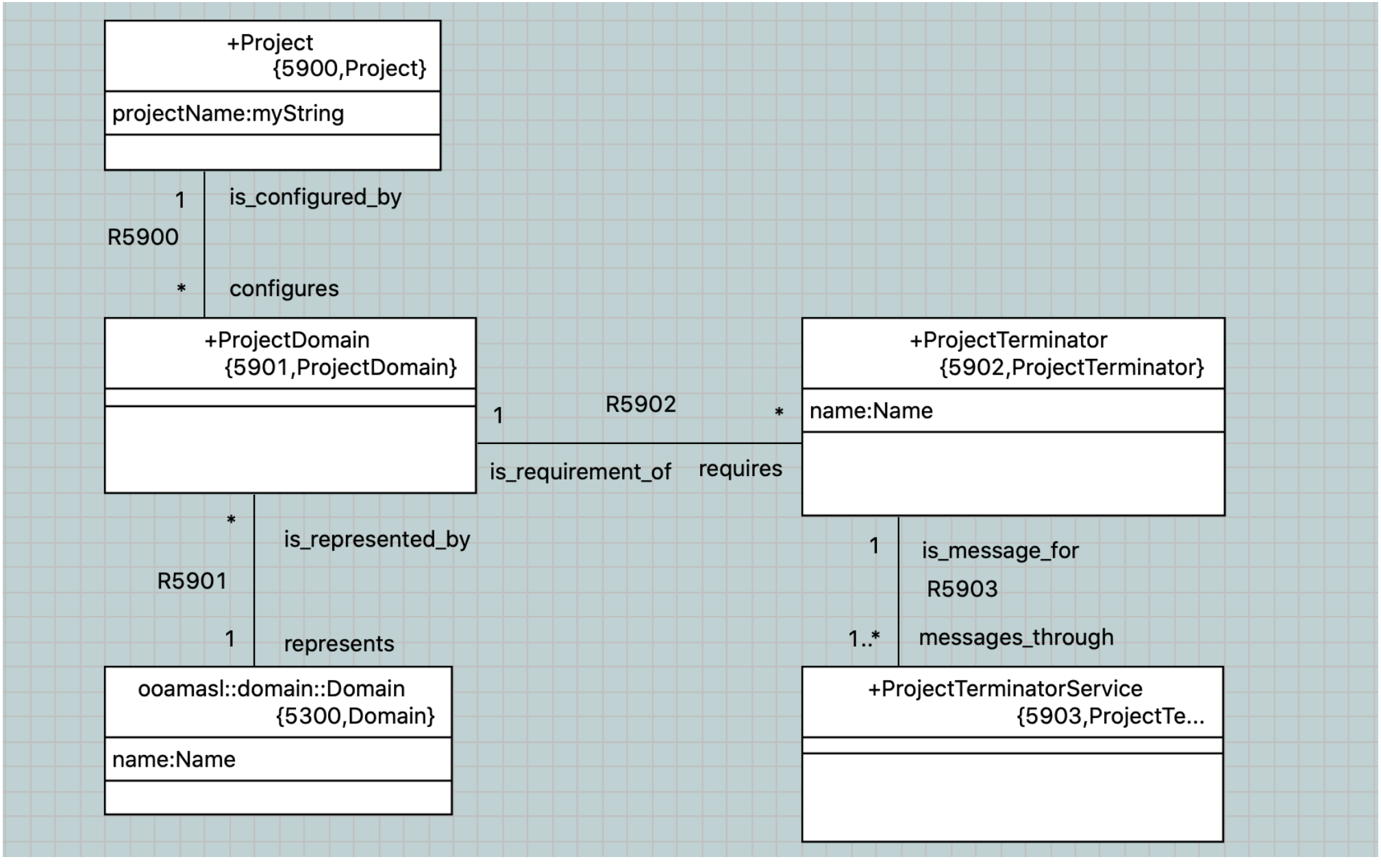
OOA of MASL: **literal**



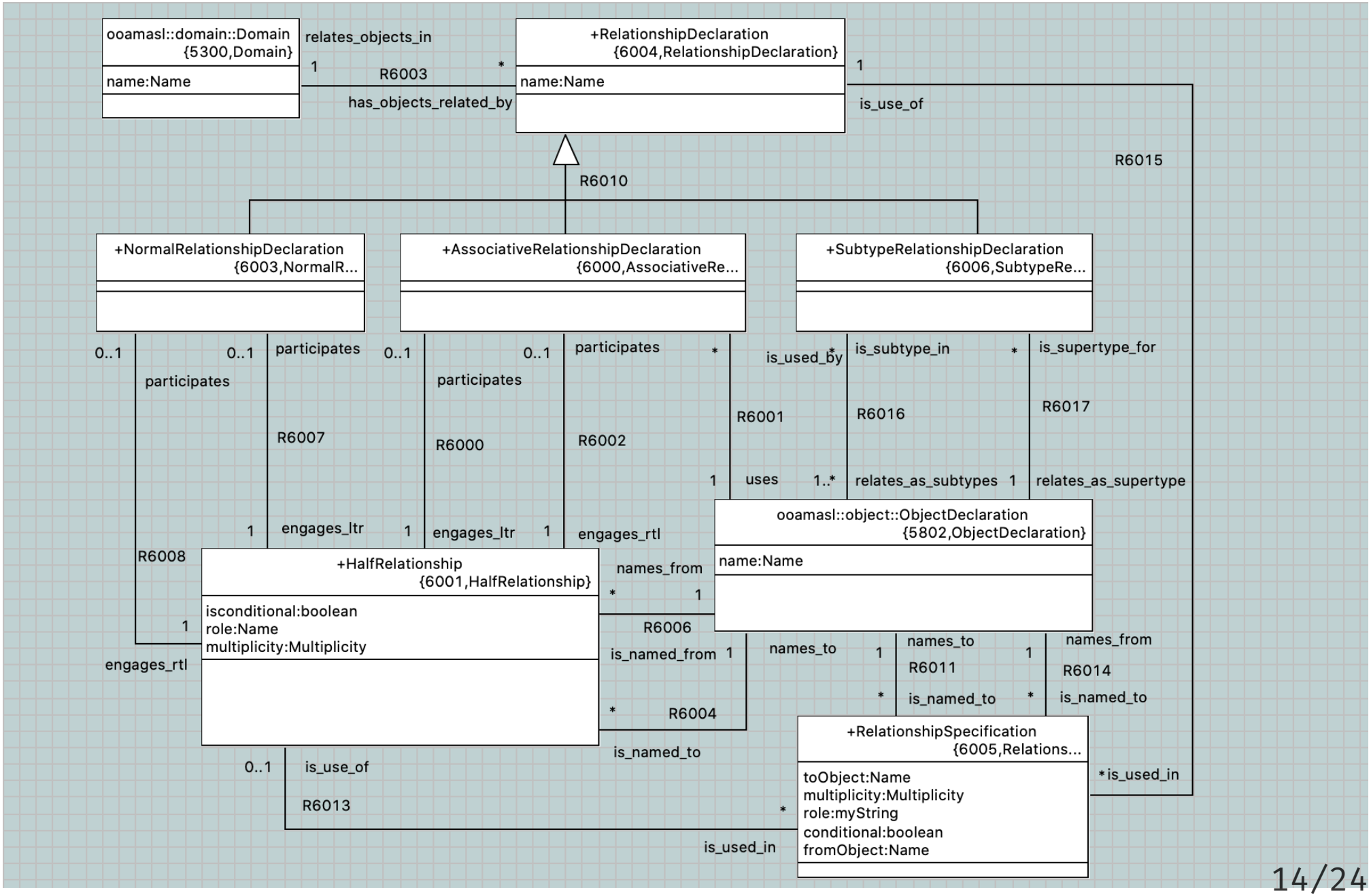
OOA of MASL: **object**



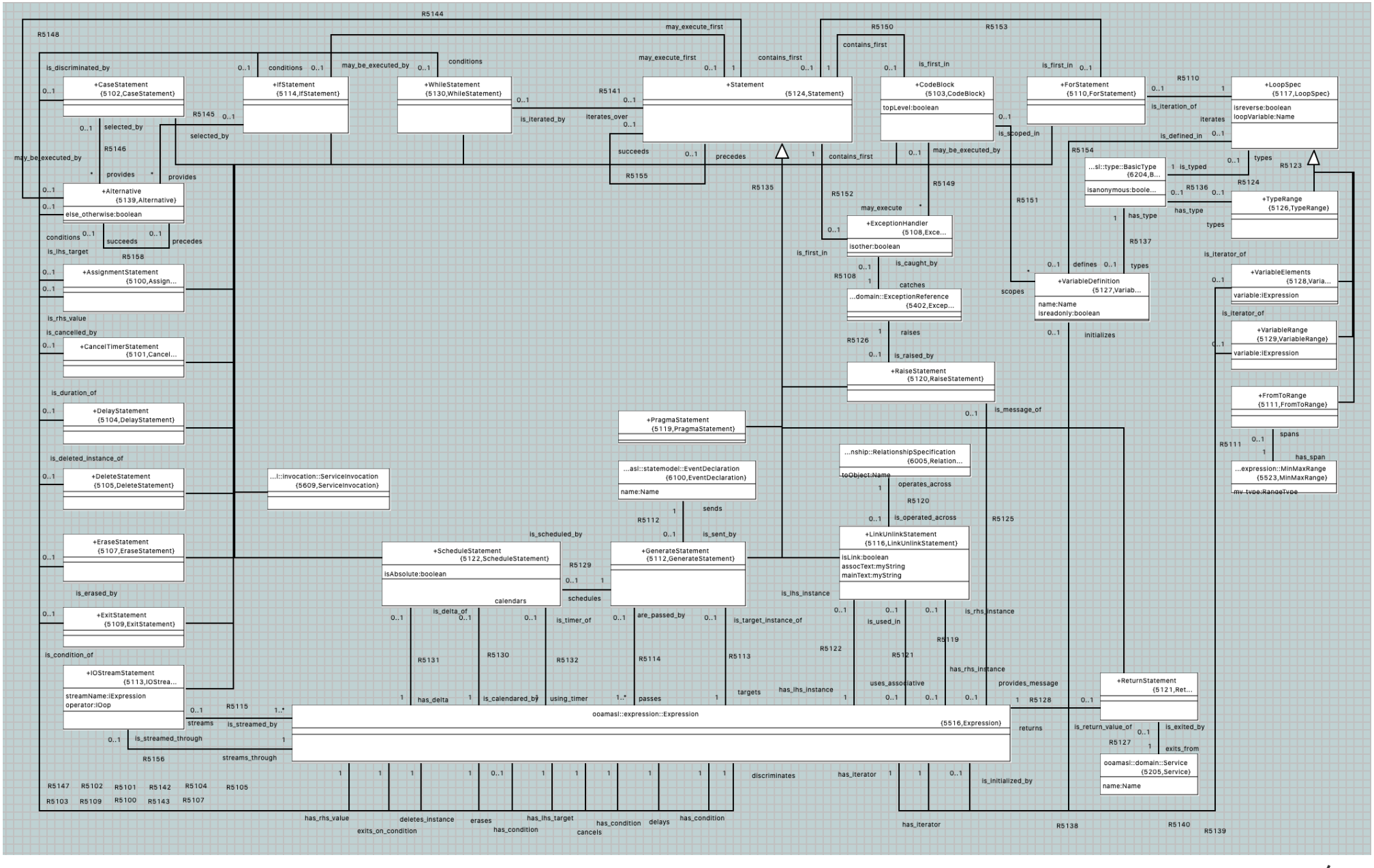
OOA of MASL: project



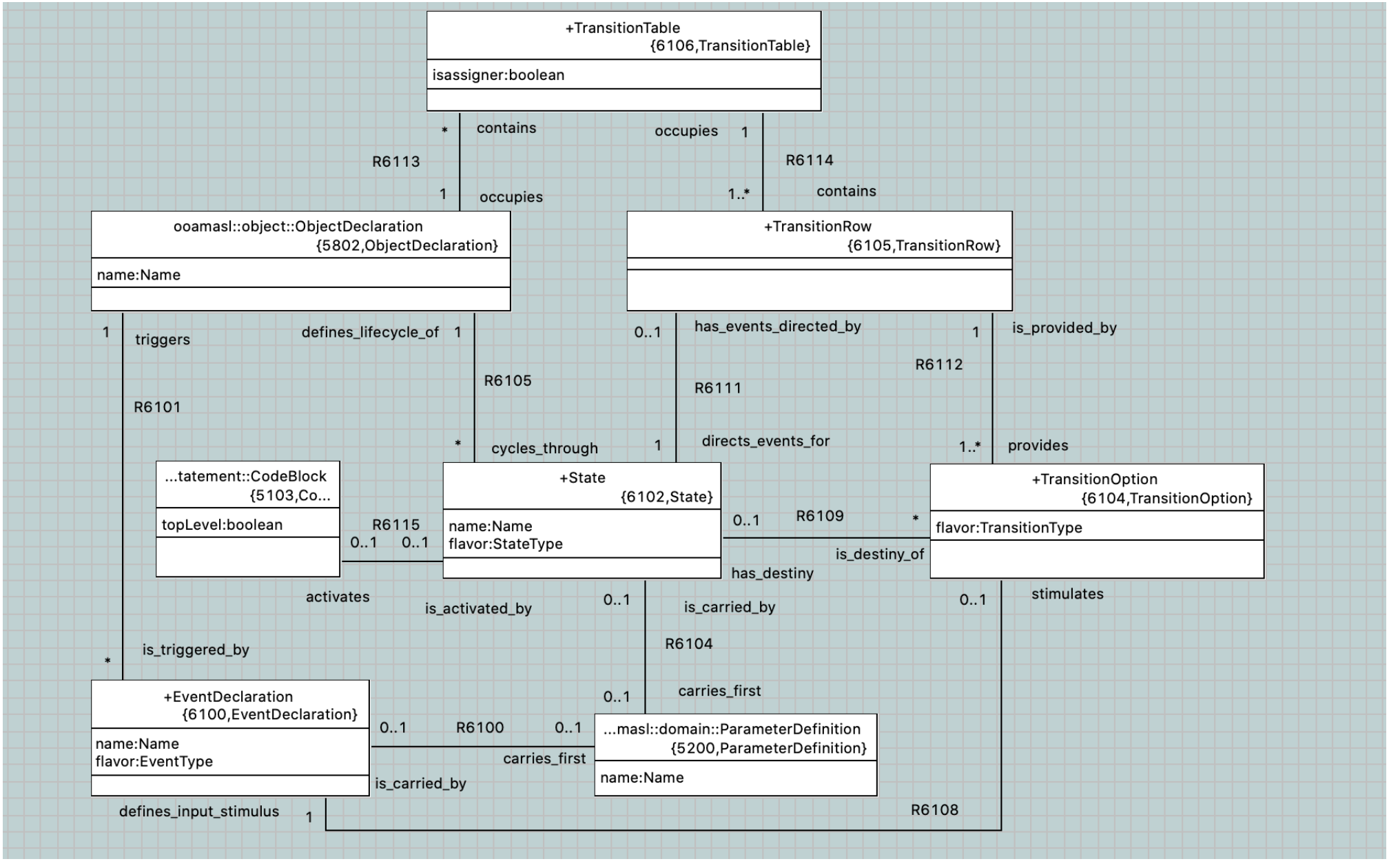
OOA of MASL: relationship



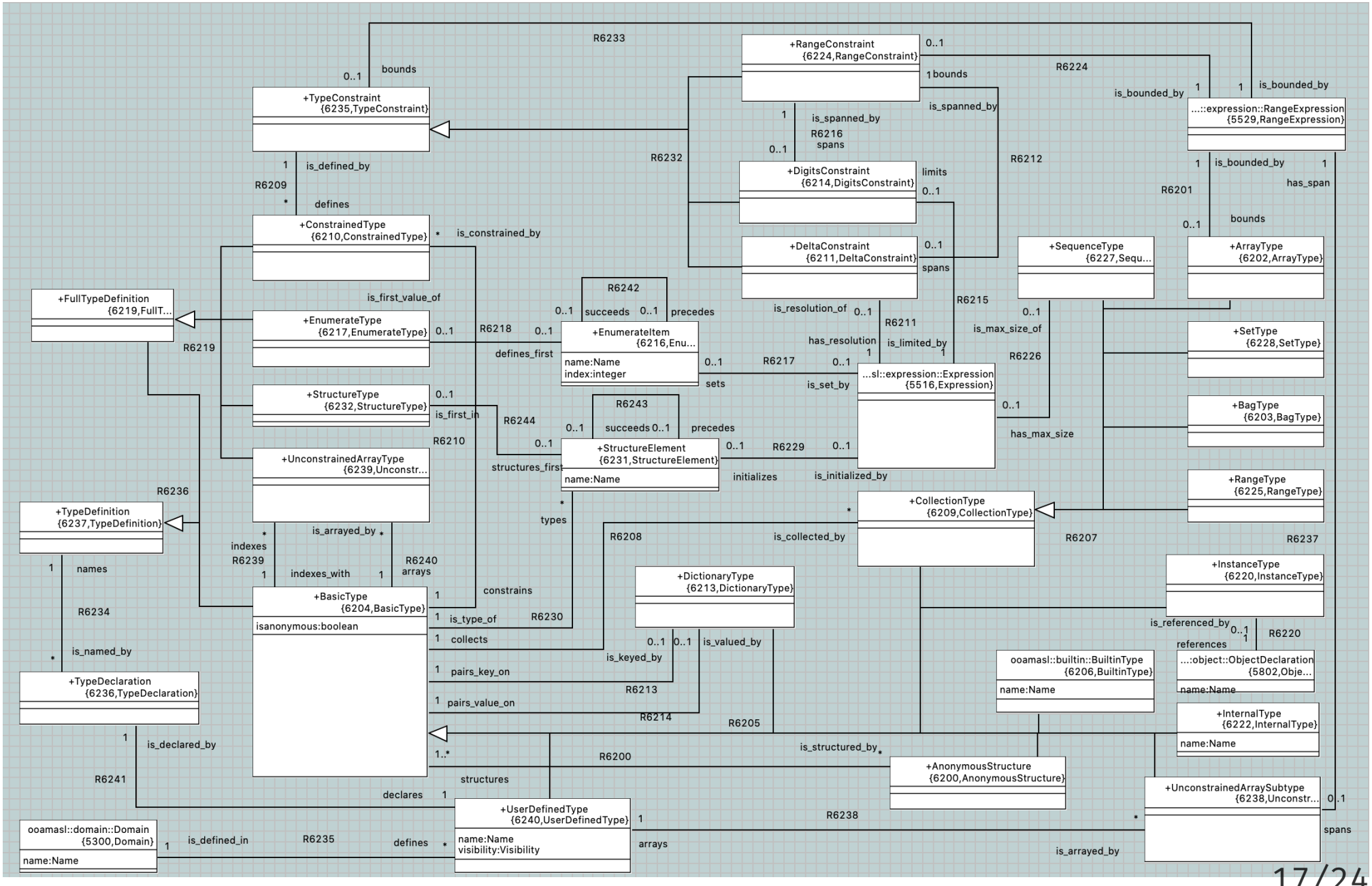
OOA of MASL: statement



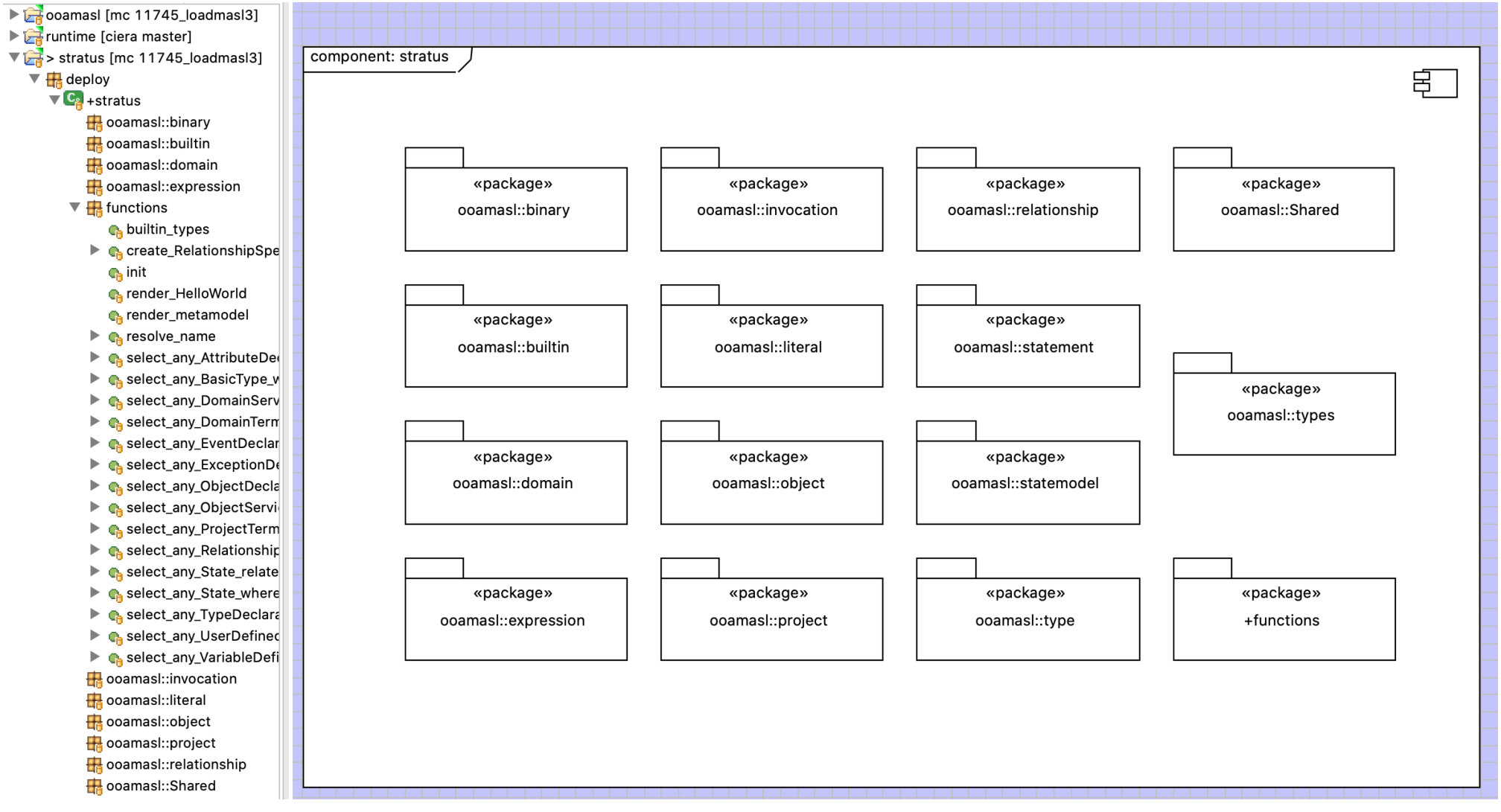
OOA of MASL: statemodel



OOA of MASL: **type**



OOA of MASL: Stratus Deployment



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	117
relationships	230
production rules	166
generated LOC (;)	51292

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`

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