OntoUML to SBVR translation pattern

For each type == Class

1. Create a new entry with the class name
2. Check the stereotype of the class

2.1: stereotype == (kind, subkind, collective, quantity, relator, category, mixin, mode, quality)

Add to the entry of this class
Concept Type = general concept

2.2: stereotype == (phase, role, phaseMixin, roleMixin)

Add to the entry of this class
Concept Type = role

For each type == Generalization (not in GenSet)

1. Find the specific {s} and the general {g} of the relation
2. Add to the entry of {s}
General concept == {g}

For each type == Generalization Set

1. Find the general {g} and all the specifics {si}
2. Add to the entry of each {si}
Necessity: each {si} is of type {g}
3. Find out whether the set is disjoint and/or complete

3.1: complete and disjoint

Add to the entry of {g}
Necessity: each {g} is exactly one {si}

3.2: complete, not disjoint

Add to the entry of {g}
Necessity: each {g} is at least one {si}

3.3: not complete, disjoint

Add to the entry of {g}
Necessity: each {g} is at most one {si}

3.4: not complete, not disjoint

Add to the entry of {g}
Possibility: it is a possibility that {g} is one or multiple {si}

For each type == Relation

1. Find out the source {s} and target {t} of the relation and their cardinalities {c;s} and {c;t}
2. Find out the type of relation

2.1: componentOf

Add to the entry of {t}
Necessity: each {t} is a component of {c;s} {s}

Add to the entry of {s}
Necessity: each {s} is composed of {c;t} {t}

2.2: characterization

Add to the entry of {t}
Necessity: each {t} characterizes exactly one {s}

Add to the entry of {s}
Necessity: each {s} is characterized by {c;t} {t}

2.3: memberOf

Add to the entry of {t}
Necessity: each {t} is a member of {c;s} {s}

Add to the entry of {s}
Necessity: each {s} has as members {c;t} {t}

2.4 mediation:

Add to the entry of {t}
Necessity: each {t} is required by {c;s} {s}

Add to the entry of {s}
Necessity: each {s} requires {c;t} {t}

2.5 material:

Add to the entry of {t}
Necessity: each {t} is connected to {c;s} {s}

Add to the entry of {s}
Necessity: each {s} is connected to {c;t} {t}

2.6 other

If the aggregation kind of the source is of type 'composite', the relation is treated as a componentOf relation. Else the relation is treated in the following way:

Add to the entry of {t}
Necessity: each {t} is associated to {c;s} {s}

Add to the entry of {s}
Necessity: each {s} is associated to {c;t} {t}

3. If the cardinality in a rule mentioned in point 2 has lower bound == 0

Replace:
Necessity: each
With:
Possibility: it is possible that a